Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
ОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
им. Ф.М. ДОСТОЕВСКОГО

А.К. Гуц

СИНТЕТИЧЕСКАЯ ВЫЧИСЛИМОСТЬ

Учебное пособие



2016

УДК 510.57:515.12
ББК 22.1
Г97

*Рекомендовано к изданию
редакционно-издательским советом ОмГУ*

Рецензенты:
д-р техн. наук, профессор В.А. Филимонов,
канд. физ.-мат. наук, доцент Н.Ф. Богаченко

Гуц, А. К.

Г97 Синтетическая вычислимость: учебное пособие / А. К. Гуц. – Омск: Изд-во ОмГУ, 2016. – 152 с.

ISBN 978-5-7779-1977-9

Излагаются элементы современной теории синтетической вычислимости. Дается представление об алгоритмах, рекурсивных функциях, интуиционизме, интуиционистской логике, конструктивной математике, реализуемости Клини, теории категорий, теории топосов, топосе реализуемости и др. Описываются эффективный топос Хайлэнда и рекурсивный топос Малри, в которых все функции $f : \mathbb{N} \to \mathbb{N}$ и $f : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ соответственно вычислимы.

Для студентов и аспирантов факультетов компьютерных наук, информационных технологий и математических факультетов.

УДК 510.57:515.12
ББК 22.1

На обложке:
К. Малевич. «Точильщик (принцип мелькания)», 1912
(Художественная галерея Йельского университета
Нью-Хейвен, США)

ISBN 978-5-7779-1977-9

Оглавление

Введение

Вычислимость чего бы то ни было означает, что это что бы то ни было может быть получено, достигнуто в результате проведенных вычислений.

Вычисление – это пошаговое выполнение некоторого набора действий, проводимых по строго установленным (и заранее оговоренным) правилам, которые могут выполняться *механически*, т. е. без осознания смысла производимых на каждом шаге действий. Осознанию, осмыслению, приданию значения подлежит только результат вычислений. Значение, смысл действия, выполняемого на каждом шаге, имело место в прошлом – до того, как это действие было кем-то придумано, обосновано, признано истинным и введено в научную практику.

Вычисления – это мысли и научные открытия поколений талантливых исследователей, оформленные как набор формальных правил. То, что с трудом когда-то далось гениям, доведено до состояния, представляющее собой действо, которое называется вычислением и которое может быть осуществлено любым, хотя и в некоторой степени подготовленным человеком, либо предоставлено для выполнения компьютеру.

Какое значение имеют вычисления для человеческого бытия? Очень важное, поскольку уже не гений, не поколения талантов размышляют над проблемой и ищут ее решение, а *формулирует* ее и находит решение с *помощью вычислений* вполне рядовой специалист. Вычисления, которые не требуют осмысления, *экономят время.* Они позволяют *быстро* найти

решение задачи или приблизиться к решению и тем самым обеспечивают благополучное существование человека.

Действительно, благополучное существование человека зависит от его способности своевременно приспособиться, подготовиться или переделать свое внешнее окружение так, чтобы желанное благополучие было обеспечено. Для осуществления своего благополучия человек вынужден совершать действия. Эти действия должны быть успешными. Задуманное необходимо заранее спланировать, продумать, его реализация должна быть гарантированной от неудачи. Гарантией успеха может быть только то, что *просчитано*, т. е. было итогом некоторых расчетов, некоторых вычислений. Умение проводить вычисления – важнейший момент в созидательной деятельности людей.

Чем совершенней, комфортнее людям хотелось бы иметь свою повседневную жизнь, тем *сложнее*, объемнее должны быть проводимые вычисления. В какой-то момент человек не стал справляться с объемом вычислений и переложил это на машину. Именно машина обеспечивает проведение вычислений механическим, т. е. бездумным образом. Вычисления, ведущиеся бездумно, т. е. без участия человека, идущие сами по себе – это реализованная мечта о скатерте-самобранке, о ведрах, шагающих самостоятельно к дому.

Как правило, вычисления совершаются по заранее составленному плану. Одна из форм такого плана – это компьютерная программа.

Таблица 1

Теория категорий	**Физика**	**Топология**	**Логика**	**Компьютер**
объект	система	многообразие	высказывание	тип данных
морфизм	процесс	кобордизм	доказательство	программа

Компьютерная программа – это план вычислений. Работающая программа – это вычисления, пребразующие один тип данных (input) в другой тип данных (output).

Программа – это отображение, подобное морфизмам в теории категорий, где морфизм один объект переводит в другой

объект. Подобная ситуация имеет место в физике, топологии и логике (см. табл. 1).

Но всё ли поддается вычислению? Можно ли всегда найти решение проблемы, производя вычисления? Всё ли вычислимо?

Окружающий нас Внешний мир воспринимается и созидается людьми шаг за шагом, минута за минутой, день за днем и т. д., иначе говоря – *во времени*. Вычисление – это познание или созидание Реальности во времени. Поэтому вычисление производится пошагово. Вычисления – способ *сопряжения* человеческого сознания с Внешним миром, т. е. с внешней Реальностью.

Но не стоит думать, что *сопряжение* человеческого сознания с Внешним миром возможно только посредством вычислений. Возможно интуитивное, вневременное сопряжения сознания с внешней Реальностью. Доказательством этого в какой-то мере является существование *невычислимого*.

Действительно, теорема Гёделя о неполноте говорит, что существуют недоказуемые высказывания. Равным образом, существуют невычислимые функции [5], [31, с. 18], недетерминированные (квантовые) процессы и некобордантные многообразия.

Однако надо понимать, что невычислимое является следствием сформулированного определения вычислимого. Вычислимое в современной *классической теории вычислимости* – это частично рекурсивные функции. После того как было дано определение таких функций, было показано, что существуют в классической арифметике и не частично рекурсивные функции. И хотя тезис Чёрча говорит, что не может быть найдено определение вычислимости, выходящее за рамки частично рекурсивных функций, – это всего лишь гипотеза, которая в силу своей формулировки не предполагает какого-либо строгого доказательства своей справедливости.

Так или иначе, но в классической арифметике не все вычислимо. И в силу этого важно понять: что же не поддается вычислениям? И что значит для человека существование

невычислимых сущностей?

Существование невычислимого Ю.Л. Ершов и В.В. Целищев связывают с существованием не допускающих алгоритмического сжатия содержания хаотических (случайных) процессов во Вселенной [23, с. 11]. В таком случае использование действительных чисел, которые ассоциируются с невычислимостью, позволяет нам моделировать Вселенную. Другими словами, то что в нашей Вселенной невычислимо, а это хаотические случайные процессы, описывается с помощью того, что сопряжено, ассоциировано с невычислимостью.

Однако, если окружающий нас внешний мир таков, что в нем есть невычислимое, то существуют ли миры, в которых *всё* вычислимо? Мы считаем наш мир реальным, естественным, данным людям изначально, независимым от людей. С этой точки зрения мир, в котором все вычислимо, является миром специально построенным, синтезированным, или синтетическим[1]. Иначе говоря, это мир, созданный людьми, т. е. искусственный. Вычислимость в нем – это *синтетическая вычислимость*. Что собой представляет синтетическая вычислимость, как она осуществляется, каковы синтетические миры?

Классическая математика, наблюдая некоторое свойство (признак), которым обладает определенная совокупность изучаемых (математических) объектов, выделяет их в особую группу объектов и говорит о существовании специфической математической структуры, которой оснащены изучаемые объекты. По Бурбаки, математическая структура

$$\langle M, P_1^{k_1}, ..., P_m^{k_m} \rangle$$

– это множество M, снабженное некоторыми специфическими отношениями (предикатами) $P_j^{k_j} \subset M^{k_j}$ $(j = 1, ..., m)$.

Таким образом, наблюдаемое свойство представляется набором предикатов. Очевидно, что не на всяком множестве мо-

[1] *Синтетический* – создаваемый путем синтеза; полученный в результате синтеза; искусственный. *Синтез* (от греч. synthesis – «соединение») – соединение (мысленное или реальное) различных элементов объекта в единое целое (систему).

жет быть задан этот набор предикатов, на котором они истинны. Иначе говоря, не все множества обладают заинтересовавшим нас свойством. Например, множество вещественных функций $\mathbb{R} \to \mathbb{R}$ не состоит только из непрерывных функций, множество непрерывных функций $\mathbb{R} \to \mathbb{R}$ не состоит только из дифференцируемых функций, множество функций $\mathbb{N} \to \mathbb{N}$ не состоит только из вычислимых функций. Это всё факты, саморазумеющиеся в мире классической математики.

Но нельзя ли построить математический мир, в котором все объекты обладают заинтересовавшим нас свойством? Хотелось бы построить такой мир, в котором все функции $\mathbb{N} \to \mathbb{N}$ вычислимы, или мир, в котором все функции $\mathbb{R} \to \mathbb{R}$ непрерывны. Другими словами, свойство, которое в классическом мире было *специфичным*, стало бы общим, *универсальным*. Такой подход называют *синтетическим* [53]. С его помощью строятся *синтетические миры*. Универсальное свойство объектов синтетического мира естественно назвать синтетическим.

Идея синтетического подхода заключается в том, что вместо построения объектов с определенными свойствами теоретико-множественными средствами их автоматически получают, вводя те или иные постулаты. Одним из примеров синтетического подхода к проблеме существования наряду с дифференцируемыми функциями недифференцируемых функций является синтетическая дифференциальная геометрия Кока–Ловера, в которой **все** функции $f : R \to R$ дифференцируемы в силу принимаемой аксиомы Ловера (см.: [77]).

Можно говорить о *внешнем* взгляде на объекты синтетического мира, который обращает внимание на специфичность тех или иных свойств, и о *внутреннем*, т. е. в рамках рассматриваемой теории, фиксирующем внимание на универсальности (синтетичности) свойства (A. Bauer, см. табл. 2).

В случае *универсального свойства вычислимости* цель построения синтетического мира состоит в том, чтобы развить теорию вычислимости искусственным образом, т. е. строится абстрактная математическая структура, в которой *все* функции и множества укомплектовываются *внутренней* структу-

рой вычисляемости. В таком синтетическом мире вычислимость является вездесущей, и в силу этого нет необходимости говорить о том, что вычисление должно вестись на машине Тьюринга, или вводить какое-либо иное понятие вычисления. В синтетическом мире вычислимые функции являются просто всеми функциями, вычислимые нумерованные множества являются всеми нумерованными множествами [53].

Таблица 2

Символ	Внешнее свойство	Внутреннее свойство
N	натуральные числа	натуральные числа
$\mathbb{R}$	*вычислимые* действительные числа	*все* действительные числа
$f : N \to N$	*вычислимые* функции	*любые* функции
$e : N \to A$	*вычислимая* нумерация	*любая* нумерация
$\top, \bot$	истинностные значения	разрешимые истинностные значения
Ω	истинностные значения топоса $\mathcal{E}\!f\!f$	истинностные значения
$\forall x$	*вычислимо* для всех x	для всех x
$\exists x$	существует вычислимое x	существует x
$P \vee \neg P$	процедура разрешимости P	P или не P

Каковы синтетические миры с синтетической вычислимостью?

В этой книге строятся синтетические миры, которые являются топосами. Топосы – это категории, для которых верны все классические аксиомы канторовской теории множеств, кроме аксиомы выбора. Самые известные топосы, представляющие миры с синтетической вычислимостью, – это эффективный топос Хайлэнда и рекурсивный топос Малри.

Эффективный топос Хайлэнда $\mathcal{E}\!f\!f$ – это математический мир, подходящий для теории синтетической вычислимости в том случае, когда под вычислимыми функциями понимаются рекурсивные функции $f : \mathbb{N}^m \to \mathbb{N}$ (функции, определенные на конечных словах). В топосе $\mathcal{E}\!f\!f$ все объекты и морфизмы обладают свойством (структурой) вычислимости.

В том случае, когда под вычислимостью понимаются вычислимые функционалы Банаха–Мазура $f : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ (функции, определенные на бесконечных словах), то математический мир, реализующий соответствующую синтетическую вычислимость, – это *рекурсивный топос Малри* RtM.

Исследователь, изучающий миры с синтетической вычислимостью, вынужден дать ответы на следующие вопросы:

1. В мире с синтетической вычислимостью, если вспомнить предположение Ю.Л. Ершова и В.В. Целищева о природе невычислимого, видимо, должны отсутствовать хаотические, или случайные, процессы – процессы, в которых события могут появиться, а могут не появиться. В нашей Вселенной таких процессов множество и их характеризуют с помощью понятия вероятности. Так что, в синтетическом мире с синтетической вычислимостью к событиям неприменимо понятия их вероятности? И, следовательно, в этом мире любое событие A либо *однозначно есть*, либо его *однозначно нет*?

2. Синтетические миры, используемые для моделирования физической реальности, должны содержать то, что именуется действительными числами. Но они, как уже говорилось, ассоциируются с невычислимостью. Поэтому, думается, в синтетических мирах с действительными числами не должно быть вероятностных величин, и отсутствие случайности должно компенсироваться внутренней интуиционистской логикой. Так ли это?

Известно, однако, что миры с интуиционистской логикой при их интерпретации (физическом моделировании) предъявляют *множество вариантов* классических миров [13], в части которых любое изучаемое событие A есть, в другой же части – его нет. Иначе говоря, вероятность заменяется на *множественность*, на *многовариантность* [10]. Отметим, что синтетические миры Хайлэнда и Малри являются топосами, внутренняя логика которых интуиционистская, а мир Хайлэнда содержит объект действительных чисел.

В книге, однако, заданные выше вопросы не рассматриваются в силу того, что они достаточно специфичны, сложны и поиск ответа на них выходит за рамки содержания книги. Основной задачей автора было представление самых различных способов построения синтетических миров с синтетической вычислимостью и изучение отдельных их свойств.

Глава 1

Вычислимость и алгоритмы

То, что вычислимо – вычисляется с помощью алгоритма. Алгоритм – это способ вычислять вычислимое. Иначе говоря, «вычислимое» и «алгоритмы» – взаимозаменяемые понятия.

1.1. Интуитивное определение алгоритма и вычислимой функции

1.1.1. Определение алгоритма

Под алгоритмом мы интуитивно понимаем перечень инструкций, разработанных для решения какой-либо задачи, с которой часто приходится иметь дело, позволяющих достаточно успешно, шаг за шагом и, в общем-то, без особых умственных усилий механически получать нечто, что мы воспринимаем как результат решения задачи, для которой данный алгоритм и был разработан.

Попробуем навести некоторую строгость в приведенное определение алгоритма.

Пусть A – некоторое множество, называемое алфавитом. Тогда множество $D \subset P(A)$ назовем совокупностью *данных*, записанных в алфавите A.

Алгоритм Al – это конечный набор инструкций (правил), по которым перерабатываются (изменяются) данные из D, удовлетворяющий условиям:

- инструкции можно применять к данным $d \in D$;
- алгоритм состоит из отдельных простых шагов; каждый шаг состоит в непосредственном применении одной из инструкций, причем совершенно ясно какой, к ситуации S_d, возникшей к этому шагу при переработке данного d, и перевод этой ситуации в новую ситуацию S_d^*;
- пошаговый процесс переработки d в d_1, d_1 в d_2, d_2 в d_3 и т. д. продолжается до тех пор, пока либо не произойдет безрезультатная остановка, либо не появится сигнал о получении «результата». При этом не исключается возможность непрекращающегося процесса переработки.

Результатом действия алгоритма являются данные $d^* \in D$, т. е. имеем отображение

$$Al : d \to d^*.$$

Данное нами определение алгоритма, несмотря на наши старания, опять-таки, не является математически строгим, поскольку использует слова, смысл которых требует уточнений, да и сами используемые термины даны без их собственного предварительного определения.

Однако существуют строгие математические формальные определения алгоритма. Это понятия рекурсивной функции, машины Тьюринга, нормального алгорифма Маркова и λ-исчисления Клини. Все эти четыре понятия алгоритма эквиваленты [13], одно влечет другое, хотя и используются при их определении совершенно разные слова и ситуации.

1.1.2. Определение вычислимой функции

Пусть дана функция

$$f : D \to D.$$

Она называется *вычислимой*, если существует алгоритм Al, с помощью которого мы вычисляем значения функции f, т. е.

$$f(d) = Al(d).$$

1.2. Рекурсивные функции Эрбрана–Гёделя–Клини

Какие функции могут быть вычислимыми? Хотелось бы иметь их описание, не связанное напрямую с понятием алгоритма.

Функцию $f : X \to Y$ будем называть *частичной*, если она определена не для каждого значения $x \in X$. Множество тех $x \in X$, для которых однозначно указано соответствующее значение функции f, называется *областью определения* функции.

Ограничимся только функциями, заданными на *множестве натуральных чисел* $\mathbb{N} = \{0, 1, ..., n, ...\}$.

1.2.1. Примитивно рекурсивные функции

Введем следующие функции:

$$s^1(x) = x + 1,$$

$$o^n(x_1, ..., x_n) = 0,$$

$$I^n_m(x_1, ..., x_n) = x_m \ (1 \leq m \leq n; \ n = 1, 2, ...),$$

называемые далее *простейшими.*

Пусть даны частичные функции $g : \mathbb{N}^n \to \mathbb{N}$ и $h : \mathbb{N}^{n+2} \to \mathbb{N}$. Частичная функция $f : \mathbb{N}^{n+1} \to \mathbb{N}$ получена из функций g, h *примитивной рекурсией,* если

$$
\begin{aligned}
&f(x_1, ..., x_n, 0) = g(x_1, ..., x_n),\\
&f(x_1, ..., x_n, y + 1) = h(x_1, ..., x_n, y, f(x_1, ..., x_n, y)).
\end{aligned} \tag{1.1}
$$

Для $n = 0$ уравнения (1.1) принимают вид ($g = const = a$)

$$
\begin{aligned}
&f(0) = a,\\
&f(x + 1) = h(x, f(x)).
\end{aligned} \tag{1.2}
$$

Как видно из уравнений (1.1), задав исходные данные $(x_1, ..., x_n, 0)$, можно шаг за шагом найти значения функции для $(x_1, ..., x_n, 1), (x_1, ..., x_n, 2), ..., (x_1, ..., x_n, m), ...$

Определение 1.1. Частичная функция $f : \mathbb{N}^{n+1} \to \mathbb{N}$ называется *примитивно рекурсивной*, если f получается из простейших функций конечным числом операций подстановки (суперпозиции) и примитивной рекурсии.

Нетрудно понять, что примитивно рекурсивные функции являются всюду определенными функциями.

1.2.2. Рекурсия и противоречивость теории

Символически задание f через примитивную рекурсию (1.1) можно записать как

$$f = \mathfrak{R}(g, h, f). \tag{1.3}$$

Рекурсия[1], как видно из ее символической записи (1.3), – это такой способ организации вычислительного процесса, при котором процедура (алгоритм) или функция в ходе выполнения составляющих ее операторов обращается сама к себе.

[1] *Recursio* (лат.) – возвращение.

В чем содержание понятия рекурсии? Что оно означает? Из определения примитивной рекурсии (1.2) видно, что для вычисления значения f для $n+1$ используется предыдущее значение $f(n)$ этой же функции, найденное шагом раньше. Таким образом, рекурсивный – значит, использующий самого себя.

Другими словами, *рекурсия* – это способ задания функции путем определения каждого из ее значений в терминах ранее определенных значений и, возможно, использования других, уже определенных функций [24, с. 40].

Рекурсивное поведение $\mathfrak{R}$ – это самоповторяющееся поведение, возвращающееся к самому себе [47, с. 196].

Очевидно, что использование рекурсии – удобный и мощный прием организации вычислений. Однако, как показал еще Тьюринг, использование рекурсии ведет к противоречиям [29, с. 51]. И это наблюдается не только на уровне наивных классических математических теорий. Известная математикам и непротиворечивая в предположении непротиворечивости арифметики[2] аксиоматическая теория множеств Цермело–Френкеля ZF в случае включения в ее язык рекурсивных функций ведет к противоречиям. Противоречивость возникает из-за того, что ZF допускает счетные множества [26].

Рекурсивные функции – основа идей конструктивной математики (см. гл. 3). Поэтому конструктивная математика совместима с абстрактной теорией множеств ZF только на конечных множествах [26, с. 379].

1.2.3. Частично рекурсивные функции

Пусть дана частичная функция $f : \mathbb{N}^n \to \mathbb{N}$. Зафиксируем данные $(x_1, ..., x_n)$. Введем функцию

$$\mathbf{M}f(x_1, ..., x_n) = \min_{y \in \mathbb{N}} \{f(x_1, ..., x_{n-1}, y) = x_n\}.$$

[2] Непротиворечивость ZF нельзя доказать, используя только аксиомы и методы этой теории, если, конечно, эта теория вообще непротиворечива. Это утверждает, как известно, теорема Гёделя о неполноте.

По существу, **M** – это операция на множестве частичных функций. Результатом является новая частичная функция с тем же числом аргументов. Назовем эту операцию *минимизацией.*

Определение 1.2. Частичная функция $f : \mathbb{N}^{n+1} \to \mathbb{N}$ называется *частично рекурсивной*, если f получается из простейших функций конечным числом операций подстановки (суперпозиции), примитивной рекурсии и минимизации.

Каждая примитивно рекурсивная функция является частично рекурсивной. Обратное неверно, поскольку в класс частично рекурсивных функций в соответствии с определением попадают частичная функция $\mathbf{M}s^1$ (см.: [13, пример 4.3]) и, например, нигде не определенная функция

$$f(x) = \min_{y \in N}\{x + 1 + y = 0\}.$$

Обратим внимание на то, что минимизация может быть организована как последовательный (алгоритмический) процесс, т. е. реализована как алгоритм.

Действительно, последовательно находим

$$f(x_1, ..., x_{n-1}, 0), f(x_1, ..., x_{n-1}, 1), ..., f(x_1, ..., x_{n-1}, y), ...$$

Наименьшее a, для которого

$$f(x_1, ..., x_{n-1}, a) = x_n,$$

– это значение для $\mathbf{M}f(x_1, ..., x_n)$.[3]

Существуют не частично рекурсивные функции. Примеры таких функций даны в [44, с. 255–256].

1.2.4. Общерекурсивные функции

Определение 1.3. Функция $f : \mathbb{N}^{n+1} \to \mathbb{N}$ называется *общерекурсивной*, или просто *рекурсивной*, если она является частично рекурсивной и всюду определенной.

[3] Возможно, что процесс нахождения не будет завершен, но это соответствует тому, что $\mathbf{M}f$ может быть истинной частичной функцией.

Множество $A \subset \mathbb{N}^m$ называется *рекурсивным* (примитивно рекурсивным), если характеристическая функция множества A

$$\chi_A(n) = \begin{cases} 1, & n \in A, \\ 0, & n \notin A \end{cases}$$

частично рекурсивна (соответственно: примитивно рекурсивна) [30, с. 87].

Предикат $R(x_1, ..., x_m)$ называется *рекурсивно перечислимым*, если его *частичная характеристическая функция*

$$\chi_R(x_1, ..., x_m) = \begin{cases} 1, & R(x_1, ..., x_m) = \top, \\ \text{не определена}, & R(x_1, ..., x_m) = \bot \end{cases}$$

частично рекурсивна [30, с. 140].

1.3. Тезис Чёрча

Теперь мы в состоянии ответить на вопрос, какие функции являются вычислимыми. Это частично рекурсивные функции. Действительно, внимательный анализ определения этих функций выявляет заложенную в это определение процедуру (алгоритм) их *вычисления*.

Напомним, что понятие вычислимости связывалось с понятием алгоритма, поэтому общепринятой является гипотеза[4], именуемая как

> **Тезис Чёрча.** *Класс алгоритмически (машинно) вычислимых частичных функций совпадает с классом всех частично рекурсивных функций.*

Частично рекурсивных функций – счетное множество [44, с. 156]. Поскольку даже одноместных функций $f : \mathbb{N} \to \mathbb{N}$ несчетное число, то в классической математике существуют *невычислимые функции*.

[4] Этот тезис не может быть доказан из-за того, что понятие алгоритмически вычислимой функции является интуитивным, т. е. отсутствует строгое определение.

Известна задача, ответ для которой нельзя вычислить. Ее называют *проблемой остановки*. Требуется ответить на вопрос: если компьютер (машина Тьюринга) начал выполнять программу, то можно ли предсказать, остановится ли выполнение программы, чтобы выдать результат, или будет работать вечно? Увы, нет такой частично рекурсивной функции, с помощью которой был бы найден ответ на данную задачу.

1.4. Машина Тьюринга

Слова «вычислимая функция» по своему смыслу предполагают, что эта функция на чем-то (или кем-то) вычисляется. Поскольку вычисление – это действие, не предполагающее осмысления того, что делается на каждом шаге, т. е. осуществляется бездумно, автоматически, то хотелось бы увидеть то устройство, которое осуществляет это машинно-бездумное вычисление.

Таким устройством является машина Тьюринга.

1.4.1. Устройство машины Тьюринга

Машина Тьюринга $\mathfrak{T}$ состоит из следующих «частей»:

1. Ленты, разбитой на конечное число ячеек.

2. Внешней памяти, принимающей одно из состояний, входящих в множество $A = \{a_0, a_1, ..., a_m\}$. Ячейки ленты находятся в одном и только в одном из состояний из множества A. Состояние a_0 называется *пустым*.

3. Внутренней памяти, принимающей одно из состояний, входящих в множество $Q = \{q_0, q_1, ..., q_n\}$. Состояние q_0 называется «стоп».

4. Головки, двигающейся вдоль ленты и считывающей содержимое ячейки, напротив которой она останавливается.

5. Механического устройства, передвигающего головку и меняющего состояния внешней и внутренней памяти. Если головка в состоянии q_j стоит напротив ячейки с номером i, то изменения состояния внутренней памяти и состояния ячейки происходят одновременно (рис. 1.1).

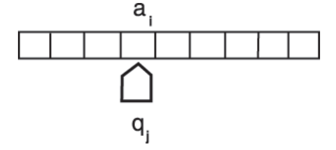


Рис. 1.1. Машина Тьюринга

Работа машины Тьюринга $\mathfrak{T}$ осуществляется посредством команд, которые выполняет механическое устройство.

Команда имеет один из следующих трех возможных видов:

$$q_s a_i \to q_t a_j, \quad q_s a_i \to q_t L, \quad q_s a_i \to q_t R,$$

где L – это движение головки влево на одну ячейку, а R – вправо. При этом всегда самое левое в записи команды $q_s \neq q_0$.

Смысл команд таков: если головка в состоянии q_s обозревает ячейку в состоянии a_i, то в первом случае она меняет свое состояние на q_t, а ячейки на a_j, во втором случае – свое состояние на q_t и сдвигается влево и, наконец, в третьем – вправо.

Конечный набор команд образует *программу* $\mathfrak{P}$.

Состояние машины $\mathfrak{T}$ – это последовательность состояний $a_{i_1}, ..., a_{i_r}$ ячеек ленты, состояние внутренней памяти g_s головки и номер k воспринимаемой (читаемой) ячейки в состоянии a_{i_k}.

Состояние машины $\mathfrak{T}$ записываем в виде

$$a_{i_1} ... a_{i_{k-1}} q_s a_{i_k} ... a_{i_r} \tag{1.4}$$

и называем *машинным словом* $\mathfrak{m}$ *в алфавите* $A \cup Q$. Символ g_s может быть самым левым, но не может быть самым правым в машинном слове, так как справа от него должно быть считываемое состояние ячейки.

Под воздействием программы происходит изменение состояния машины, сопровождающееся переделкой исходного машинного слова

$$\mathfrak{m} \to \mathfrak{m}^{(1)} \to ... \to \mathfrak{m}^{(p)}.$$

1.4.2. Вычисление функций на машине Тьюринга

Пусть $\mathfrak{m}$ машинное слово в алфавите $A \cup Q$ машины $\mathfrak{T}$. Вычеркнем из слова $\mathfrak{m}$ букву a_0 и буквы из Q. Получаем редуцированное слово $[\mathfrak{m}]$.

Редуцированное слово $[\mathfrak{m}]$ перерабатывается машиной $\mathfrak{T}$ в (редуцированное) слово $\mathfrak{b}$, если для некоторой программы $\mathfrak{P}$ и некоторого p

$$q_1 a_0 [\mathfrak{m}] \to (q_1 a_0 [\mathfrak{m}])^{(1)} \to ... \to (q_1 a_0 [\mathfrak{m}])^{(p)}$$

и

$$q_0 \in (q_1 a_0 [\mathfrak{m}])^{(p)}, \quad \mathfrak{b} = [(q_1 a_0 [\mathfrak{m}])^{(p)}].$$

Символически факт переработки слова $[\mathfrak{m}]$ записываем в виде

$$\mathfrak{b} = \mathfrak{P}([\mathfrak{m})]).$$

Если ни для какого натурального p не выполняется условие

$$q_0 \in (q_1 a_0 [\mathfrak{m}])^{(p)},$$

то говорят, что машина *неприменима* к слову $[\mathfrak{m}]$. В этом случае выражение $\mathfrak{P}([\mathfrak{m})]$ является *неопределенным.*

Пусть $f(\mathfrak{r})$ функция, определенная на словах редуцированного алфавита $[A] = \{a_1, ..., a_m\}$. Если для каждого слова $\mathfrak{r}$ в этом алфавите

$$f(\mathfrak{r}) = \mathfrak{P}(\mathfrak{r}),$$

то говорим, что машина $\mathfrak{T}$ *вычисляет функцию* f.

Машинное предложение – это выражение вида

$$a_0 \mathfrak{r}_1 a_0 \mathfrak{r}_2 a_0 ... a_0 \mathfrak{r}_t,$$

где $\mathfrak{r}_i$ – слова в словаре $[A]$.

Программа $\mathfrak{P}$ перерабатывает машинное предложение в редуцированное слово $\mathfrak{b}$, если

$$q_1 a_0 \mathfrak{r}_1 a_0 \mathfrak{r}_2 a_0 ... a_0 \mathfrak{r}_t \to (q_1 a_0 \mathfrak{r}_1 a_0 \mathfrak{r}_2 a_0 ... a_0 \mathfrak{r}_t)^{(1)} \to ...$$

$$... \to (q_1 a_0 \mathfrak{r}_1 a_0 \mathfrak{r}_2 a_0 ... a_0 \mathfrak{r}_t)^{(p)}$$

и

$$q_0 \in (q_1 a_0 \mathfrak{r}_1 a_0 \mathfrak{r}_2 a_0 ... a_0 \mathfrak{r}_t)^{(p)}, \; \mathfrak{b} = [(q_1 a_0 \mathfrak{r}_1 a_0 \mathfrak{r}_2 a_0 ... a_0 \mathfrak{r}_t])^{(p)}].$$

Символически

$$\mathfrak{b} = \mathfrak{P}(\mathfrak{r}_1, \mathfrak{r}_2, ..., \mathfrak{r}_t).$$

Если $f(\mathfrak{r}_1, ..., \mathfrak{r}_t)$ – *частично словарная функция*, т. е. функция, определенная на словах некоторого редуцированного алфавита $[A] = \{a_1, ..., a_m\}$, обладает свойством

$$f(\mathfrak{r}_1, ..., \mathfrak{r}_t) = \mathfrak{P}(\mathfrak{r}_1, \mathfrak{r}_2, ..., \mathfrak{r}_t),$$

то говорим, что *она вычислима на машине* $\mathfrak{T}$.

Теорема 1.1. *Все частично словарные функции, вычислимые на машине Тьюринга, являются частично рекурсивными* [30, с. 228]. ■

Следующая теорема подтверждает, что машина Тьюринга и есть то устройство, которое производит вычисления.

Теорема 1.2. *Для любой частично рекурсивной функции существует вычисляющая ее машина Тьюринга* [30, с. 230]. ■

1.4.3. Тезис Тьюринга

Вспоминая тезис Чёрча, естественно провести следующее рассуждение: если вычислимые функции – это частично рекурсивные, а последние вычисляются на машинах Тьюринга, то не содержится ли в этом рассуждении ответ на то, что такое вычислимая функция? Утверждение, которое хотелось бы сделать, но которое не поддерживается строгим доказательством, – это

Тезис Тьюринга. *Все вычислимые частичные функции вычисляются на машинах Тьюринга.*

1.4.4. Универсальная машина Тьюринга

Машина Тьюринга называется *универсальной*, если она может при определенных начальных входных данных вычислить любую функцию, которая вычислима на какой-либо машине Тьюринга.

Иначе говоря, с учетом тезиса Тьюринга можно сказать, что универсальная машина Тьюринга способна вычислить любую вычислимую функцию. Доказано, что универсальная машина Тьюринга существует.

1.5. Два типа вычислимости

Основным понятием теории вычислимости является вычислимая функция, т. е. функция, значения которой могут быть получены посредством вычисления.

Выделяют *два основных типа вычислимости.*

1.5.1. Вычислимые функции первого типа

К *первому типу* относят функции, которые определены на конечных словах, построенных с помощью конечного алфавита. Ясно, что этот тип эквивалентен изучению функций вида $f : \mathbb{N}^m \to \mathbb{N}$, где $\mathbb{N}$ – множество натуральных чисел.

Вычислимыми функциями первого типа считаются частично рекурсивные функции.

1.5.2. Вычислимые функции второго типа

Ко *второму типу* относят функции, определенные на бесконечных словах. Следовательно, в этом случае изучаются функции, заданные на последовательностях натуральных чисел, т. е. функции вида $f : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$.

Здесь допускаются два подхода.

Первый подход. Пространство Бэра. При первом подходе *вычислимыми функциями второго типа* считаются непрерывные функции $f : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$, когда $\mathbb{N}$ снабжается дискретной топологией, а $\mathbb{N}^{\mathbb{N}}$, которое по сути является произведением $\prod_{i=1}^{\infty} \mathbb{N}$, снабжается топологией произведения Тихонова $\mathcal{T}_{\prod}$. Таким образом, построенное топологическое пространство $< \mathbb{N}^{\mathbb{N}}, \mathcal{T}_{\prod} >$ именуют *пространством Бэра*.

Второй подход. Вычислимые функционалы Банаха–Бэра. При втором подходе функция $f : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ называется *вычислимой по Банаху–Мазуру*, если для любой двуместной общерекурсивной функции $g : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ функция $x \mapsto f(g(x, -)) : \mathbb{N} \to \mathbb{N}$ является общерекурсивной.

Заметим, что функция вида $f : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ – это функция, определенная на одноместных функциях $g \in \mathbb{N}^{\mathbb{N}}$. Считая такие функции вычислимыми и предполагая, что все одноместные вычислимые функции нумеруются натуральными числами, можно представить все вычислимые одноместные функции как всевозможные двуместные вычислимые функции $g(n, m)$, где n – это номер вычислимой функции, а m – ее аргумент. В таком случае, вычислимые по Банаху–Мазуру функции – это функционалы, определенные на функциях из $\mathbb{N}^{\mathbb{N}}$ и переводящие вычислимые функции в вычислимые:

$$f(g(n, m)) = h(n), \quad h - \text{вычислимая}.$$

Или, поскольку функция $g \in \mathbb{N}^{\mathbb{N}}$ суть последовательность, то вычислимые по Банаху–Мазуру функции – это функционалы, переводящие вычислимые последовательности в вычислимые последовательности.

1.6. Вычислимые функции действительного переменного

Действительное число называется *вычислимым*, если существует алгоритм, находящий сколь угодно точные рациональ-

ные приближения к этому числу[5].

Следовательно, например, действительное число $x \in \mathbb{R}$ *вычислимо*, если существуют две общерекурсивные функции $f, g : \mathbb{N} \to \mathbb{N}$ такие, что при всех n

$$\left| x - \frac{f(n) - g(n)}{n+1} \right| < \frac{1}{n+1}.$$

Вычислимые числа включают в себя все алгебраические и некоторые трансцендентные числа. Существуют невычислимые действительные числа.

Последовательность $\{x_n\}$ действительных чисел называется *вычислимой*, если существуют две общерекурсивные функции $f, g : \mathbb{N}^2 \to \mathbb{N}$ такие, что при всех n, k

$$\left| x_k - \frac{f(k,n) - g(k,n)}{n+1} \right| < \frac{1}{n+1}.$$

Множество C вычислимых последовательностей вычислимых действительных чисел – это множество вычислимых последовательностей действительных чисел.

Функция действительного переменного f называется *вычислимой по Банаху–Мазуру* (множество таких функций обозначим $\mathbf{R}$), если для любой последовательности $\{x_n\} \in C$ последовательность $\{f(x_n)\} \in C$.

1.7. Квантовая вычислимость

Под классическими вычислениями, определяемыми выше, мы понимали процедуры, полностью определенные и детерминированные. В реальности мы имеем дело с физическими процессами, которым в большой степени присуща неопределенность и недетерминированность.

Это и определяет появление классически невычислимых функций [23, с. 11]. Поэтому, создавая теории, где всё вычислимо, можно сохранять определенность и детерминированность в понимании вычислимости и строить искусственные миры с

[5] Приведенное определение было дано Борелем в начале XX века.

синтетической вычислимостью. Можно также создавать недетерминированные (вероятностные) машины Тьюринга и надеяться, что на них всё удастся вычислить[6]. Но можно обратиться к квантовой механике, предметом которой являются квантово-механические системы с внутренне присущими им неопределенностью и недетерминированностью и которая весьма успешно описывает реальные неопределенные и недетерминированные физические процессы.

Другими словами, можно строить теорию *квантовой вычислимости* [12], [14], [18, с. 15]. Многие специалисты по квантовым компьютерам считают, что поскольку Вселенная описывается в терминах квантовой механики, то Вселенная представляет собой квантовый компьютер, который сам себя вычисляет [29]. Следовательно, в мире, в котором мы существуем, всё квантово вычислимо: Вселенная – это мир с синтетической квантовой вычислимостью.

1.8. Алгоритмы без первого шага

«Во всех имеющихся теориях вычислимости (как детерминированной, так и недетерминированной) всякое конкретное вычисление непременно имеет первый шаг выполнения. Тем самым любой гипотетический не имеющий начала процесс автоматически оказывается за рамками возможности его моделирования средствами этих теорий» [2].

Если время – это вычислительный процесс, как полагает А.М. Анисов [3], и переход от настоящего к прошлому и будущему – это результат вычислений, производимых самой природой, то невычислимого быть не должно. И поскольку в классической теории вычислений невычислимое существует, то либо время не сводится к вычислениям, либо классическая теория вычислений нуждается в совершенствовании.

Мир, в котором все вычислимо в классическом смысле, не нуждается в алгоритмах без первого шага.

---

[6] Известно, что этот путь не дает ничего нового по сравнению с классической машиной Тьюринга [35, с. 89].

Глава 2

Конструктивная математика

Мысль о том, что следует иметь дело только с тем, что вычисляется, приводит нас к утверждению, что коль скоро в математике очень часто доказывается истинность некоторого утверждения, и при этом процедура доказательства часто использует принципы, не связанные со способом проверки реального существования того, существование чего они утверждают, то необходимо пересмотреть сами принципы получения новых знаний в математике.

Не следует ли признать, что реально существует только то, что можно построить или, по крайней мере, предъявить алгоритм построения этого нечто, вопрос о существовании или несуществовании которого является важной научной или практической задачей?

Если вы согласны с необходимостью принятия такого положения, то вы становитесь сторонником *конструктивной логики* и основанной на ней *конструктивной математики*.

Конструктивное направление в математике в СССР и в России складывалось под влиянием работ А.А. Маркова-младшего и его учеников. Толчком к развитию конструктивной ма-

тематики стала критика методов классической математики, развернутая в начале XX века Брауэром и Германом Вейлем.

2.1. Интуиционизм

Идеям конструктивной математики предшествовал интуиционизм Л.Э.Я. Брауэра.

«Свое начало интуиционизм берет в работах Канта, который пытался вывести математическое понятие из чистого созерцания пространства и времени и настаивал на конструктивном характере производных математических объектов. Голландский математик Л. Брауэр положил идеи интуиционизма в основу своей программы перестройки классической математики» [41].

Брауэр ограничивает область действия классической логики, которую Кант принимал в качестве абсолютного и универсального инструмента мышления. Классическая логика ограничивается Брауэром за счет отказа от использования закона исключенного третьего применительно к бесконечным множествам и за счет существенного изменения смысла таких понятий, как «все» и «существует». Согласно Брауэру, мы можем утверждать нечто о всех элементах множества в целом только в том случае, если имеем конструктивную процедуру обоснования этого утверждения для каждого элемента этого множества в отдельности. Общая установка Брауэра состоит в том, чтобы использовать логику только в тех границах, в которых она не выводит математическое рассуждение за сферу прямого конструктивного оправдания.

«Только после того, полагает Брауэр, как математику станут считать конструктивной деятельностью, основанной на интуиции времени, которая хотя и может быть применена к внешнему миру, но ни по своему происхождению, ни по своим методам не зависит от него, а критерий истинности математического утверждения будет ограничен самой математической деятельностью, математика достигнет полной автономии и самооправдания. Непосредственным следствием такого ра-

дикального изменения становится то, что для произвольного математического утверждения p допускавшиеся классической логикой две альтернативы, истина или ложь, заменяются следующими тремя:

• истинно то утверждение, которое доказано;

• ложно то утверждение, которое, как доказано, абсурдно (противоречиво);

• не истинно и не ложно то утверждение, истинность или абсурдность которого не доказаны, или неизвестен конечный алгоритм доказательства его истинности или ложности.

Случай, когда не доказаны ни истинность, ни абсурдность утверждения, но алгоритм такого доказательства известен, сводится, как очевидно, к первым двум случаям» [39].

«Математика, по Брауэру, должна быть обоснована исключительно на первичных математических интуициях, логические же схемы должны считаться приемлемыми лишь в той мере, в которой они согласуются с этими интуициями. Брауэр возражал против попыток строгого формального определения допустимой логики и принципов конструирования, полагая, что то и другое определяется на основе непосредственной интуиции в самом акте математического мышления и в зависимости от его содержания... Строгость математики, по Брауэру, обеспечивается не на уровне языка и символов, а исключительно на уровне интуитивной ясности рассуждения. Строгое математическое рассуждение, подчеркивал Брауэр, реализуется не на бумаге, а в голове математика» [41, с. 209–210].

Брауэр считал, что «множество натуральных чисел должно было уступить место потенциально бесконечному натуральному ряду, бесконечному в своем развитии, в возможности построения все новых и новых натуральных чисел» [28, с. 11].

Одно из заключений, к которому пришел Брауэр и которое известно как *принцип Брауэра*, – это утверждение, что каждая функция $\mathbb{R} \to \mathbb{R}$ должна быть непрерывной [60].

Идеи и понятия, предложенные Брауэром, были, возможно, интуитивно понятны самому автору, но не понятны подавляющему большинству современных ему математиков. Инту-

ционистская математика в изложении Брауэра представляла собой туманно-расплывчатую теорию. Тем не менее его усилия не остались незамеченными.

Последователи Брауэра отошли от его расплывчатых формулировок и предпочли основываться на формализованных логических принципах интуиционизма и использовать более строгие математические определения и критерии. Развивая идеи интуиционизма, Гейтинг построил формальную интуиционистскую логику, а Клини – семантику интуиционистской логики. Но при этом создавалось принципиально отличное от интуиционистских идей Брауэра конструктивное направление в математике.

Переход от интуиционизма Брауэра к *конструктивной математике* произошел благодаря тому, что в работах Алана Тьюринга, Эмилля Поста и А.А. Маркова-младшего было найдено строгое определение понятия алгоритма. Благодаря этому конструктивное направление в математике, конструктивная математика кардинально отличается от интуиционистской математики Брауэра.

2.2. Интуиционизм Канта

«В философии Канта мы находим старую форму интуиционизма, ныне почти отброшенную, в которой время и пространство считаются априорными формами чувственности, прирожденными человеческому разуму. Для Канта аксиомы арифметики и геометрии – априорные синтетические суждения, т. е. суждения, независимые от опыта и недоказуемые аналитически; именно этим объяснялась их аподиктическая точность в мире опыта и в абстракции. Поэтому для Канта возможность экспериментального опровержения арифметических и геометрических законов не только исключалась твердым убеждением в их истинности, но и была просто немыслима» [39].

Открытие геометрии Лобачевского поколебало систему интуиционизма Канта, поскольку возникла проблема признания априорности одного из двух взаимоисключающих геометриче-

ских законов – евклидова или Лобачевского. Сам Лобачевский пытался сделать это чисто экспериментальным путем, измеряя расстояние между звездами. Но эти геометрические открытия не касаются утверждения Канта об априорности времени, которое он связывал с арифметикой.

«Диаметрально противоположна точка зрения формализма, который утверждает, что человеческий разум имеет в своем распоряжении образов прямых линий или чисел, скажем, не более десяти, и поэтому источник этих математических объектов находится не в нашем представлении природы, а в самой природе... Для формалиста, следовательно, математическая точность сводится к созданию метода вывода одних отношений об объектах из других и не зависит от значения, которое можно приписать этим отношениям или связываемым ими объектам» [39].

2.3. Интуиционистская логика Гейтинга

С точки зрения последовательного сторонника интуиционизма Брауэра невозможно формализовать, т. е. представить в аксиоматическом виде, интуиционистскую логику.

Гейтинг, ученик Брауэра, писал: «построить формальную систему, которая бы была эквивалентна интуиционистской математике, в принципе невозможно... никогда нельзя математически строго доказать, что данная аксиоматическая система охватывает все правильные методы доказательств» (цит. по: [9, с. 191]).

Тем не менее, А.Н. Колмогоров (1925, [27]), В.И. Гливенко (1928, [8]), Гейтинг (1930, [71]), Генцен (1934, [70]) предложили формальные варианты изложения интуиционистской логики Брауэра.

Система Гейтинга считается первой формальной системой IL аксиом интуиционистской логики. Она состоит из сле-

дующих схем аксиом:

$$\begin{aligned}
&\mathcal{A} \Rightarrow (\mathcal{A} \wedge \mathcal{A}),\\
&(\mathcal{A} \wedge \mathcal{B}) \Rightarrow (\mathcal{B} \wedge \mathcal{A}),\\
&(\mathcal{A} \Rightarrow \mathcal{B}) \Rightarrow ((\mathcal{A} \wedge \mathcal{C}) \Rightarrow (\mathcal{B} \wedge \mathcal{C})),\\
&((\mathcal{A} \Rightarrow \mathcal{B}) \wedge (\mathcal{B} \Rightarrow \mathcal{C})) \Rightarrow (\mathcal{A} \Rightarrow \mathcal{C}),\\
&\mathcal{A} \Rightarrow (\mathcal{B} \Rightarrow \mathcal{A}),\\
&(\mathcal{A} \wedge (\mathcal{A} \Rightarrow \mathcal{B})) \Rightarrow \mathcal{B},\\
&\mathcal{A} \Rightarrow (\mathcal{A} \vee \mathcal{B}),\\
&(\mathcal{A} \vee \mathcal{B}) \Rightarrow (\mathcal{B} \vee \mathcal{A}),\\
&((\mathcal{A} \Rightarrow \mathcal{C}) \wedge (\mathcal{B} \Rightarrow \mathcal{C})) \Rightarrow ((\mathcal{A} \wedge \mathcal{B}) \Rightarrow \mathcal{C}),\\
&\neg\mathcal{A} \Rightarrow (\mathcal{A} \Rightarrow \mathcal{B}),\\
&((\mathcal{A} \Rightarrow \mathcal{B}) \wedge (\mathcal{A} \Rightarrow \neg\mathcal{B})) \Rightarrow \neg\mathcal{A}.
\end{aligned}$$

Если к этой системе добавить закон исключенного третьего, т. е. аксиому

$$\mathcal{A} \vee \neg\mathcal{A},$$

то получим систему CL аксиом классической двузначной логики.

Правило вывода для обеих логик IL и CL одно и то же:

$$\frac{\mathcal{A}, \mathcal{A} \Rightarrow \mathcal{B}}{\mathcal{B}} \quad (\text{modus ponens}).$$

Интуиционисты рассматривают систему IL как некоторый инструмент, который полезен для исследования установления различий между классической и интуиционистской математиками.

Формальная теория (логическое исчисление) является дедуктивной системой, формулы в ней доказываются, а установление их истинности требует наполнения их смыслом, значением, что делается посредством интерпретации. В рамках конструктивной математики эта интерпретация должна предполагать эффективный (конструктивный) способ проверки истинности рассматриваемых формул, утверждений формаль-

ной теории. Одним из таких способов является *теория рекурсивной реализуемости* формул, предложенная Клини (см. гл. 3).

Поскольку математика, как правило, строится как семантическая система, сопровождаемая классической двузначной логикой CL, то конструктивная математика должна развиваться, опираясь на собственную *конструктивную логику*[1], в которой установление истинности утверждений должно представлять собой конструктивный процесс.

2.4. Конструктивная математика Маркова

Основная цель конструктивной математики – избавиться от традиционных для классической математики *неэффективных* методов, посредством которых доказывается существование того или иного объекта с заданным свойством, но при этом не показывается, как такой объект с этим свойством может быть построен.

Критика неэффективных методов, которые допускаются в классической математике, привела Бауэра к тому, что он явно указал, что неэффективные методы появляются отчасти от того, что классическая математика базируется на классической двузначной логике, одной их аксиом которой является закон исключенного третьего. В пределах логики высказываний закон исключенного третьего – это единственный неэффективный логический принцип [37, с. 49]. Брауэр предложил отказаться от использования этого закона. Как результат, возникла интуиционистская логика, формализованную теорию которой первым построил Гейтинг [71]. Она содержит логические принципы конструктивистской математики.

Допустимые методы должны быть только эффективными,

[1] Польские математики предложили вариант конструктивной математики, который сохраняет классическую логику и опирается на конструктивные методы.

т. е. существование объекта с заданным свойством считается доказанным, если дана конструкция, способ, алгоритм его построения. Отсюда название новой математики – *конструктивная математика*.

Конструкции, с помощью которых устанавливается истинность тех или иных утверждений и существование объектов с заданными свойствами, должны основываться на особых конструктивных процессах, представляющих собой алгоритмы. Возможными алгоритмами могут быть нормальные алгорифмы Маркова (см. § 2.5).

2.4.1. Конструктивные натуральные числа

Марков начинает построение конструктивной математики с демонстрации конструктивности натуральных чисел. Например, число 6 – это *конструктивный процесс*, заключающийся в начертании шести черточек:

$$||||||$$

Ряды вертикальных черточек вроде данного рисунка, включая и «пустой» ряд, в состав которого не входит ни одна черточка (его можно представить себе в виде чистого листа бумаги), называются *натуральными числами*.

«Введенные таким образом натуральные числа суть конструктивные объекты.

Натуральное число, представленное нашим рисунком, называется "шесть"и иногда изображается арабской цифрой "6". Натуральное число, представленное пустым рядом черточек, называется нулем и изображается цифрой "0" » [33, с. 23].

Конструктивный процесс, как видим, в данном случае состоит в написании некоторого элементарного знака, в приписывании к нему справа другого элементарного знака, затем еще одного и т. д. до конца процесса. Результирующий конструктивный объект будет в этом случае рядом элементарных знаков.

2.4.2. Потенциальная осуществимость конструктивного процесса. Конструктивные объекты

Очевидно, написать все знаки, нарисовать все натуральные числа не в наших силах – у нас для этого не хватит либо времени, либо бумаги, либо краски для рисования черточек. Чтобы справиться с данным обстоятельством в конструктивной математике принимается *принцип потенциальной осуществимости*, полагающий, что нам следует рассуждать так, будто указанных выше препятствий для осуществления конструктивного процесса не существует; мы дожны поступать так, «как если бы в каждый момент в нашем распоряжении были и пространство, и время, и материал, потребные для осуществления очередного шага рассматриваемого конструктивного процесса» [33, с. 34].

Под *существованием объекта* с определенными свойствами в конструктивной математике понимается потенциальная осуществимость конструктивного процесса, результатом которого является объект с данными свойствами. Объект, существование которого утверждается, является тем самым *конструктивным объектом*, поскольку он представляет собой результат конструктивного процесса.

Построением конструктивного объекта называется всякий конструктивный процесс, результат которого одинаков с этим объектом. Конструктивные объекты строятся по определенным правилам из некоторых элементарных объектов, т. е. неразложимых в процессе всего построения, шаг за шагом.

Математика – абстрактная наука, она имеет дело со знаками, точнее, со словами, т. е. с конечными последовательностями знаков. Следовательно, конструктивная математика в качестве главного типа изучаемых конструктивных объектов имеет дело с частным случаем знаковых комплексов – словами в том или ином алфавите. Натуральные числа могут быть охарактеризованы как слова в алфавите, единственной буквой которого является вертикальная черточка $|$. В качестве шага конструктивного процесса может при этом фигурировать не

только написание нового элементарного знака, но и уничтожение старого знака, а также сравнение на предмет их одинаковости или различия.

Мы видим, что конструктивный процесс – это алгоритм, осуществляющий то или иное вычисление.

> *Алгорифм есть предписание, однозначно определяющее ход некоторых конструктивных процессов* (А. Марков, [33, с. 135]).

2.4.3. Конструктивная логика

При построении семантики конструктивной логики нужно не просто говорить об истинности логических формул (утверждений), но и уметь это *устанавливать* посредством некоторых конструктивных процессов.

Для логики высказываний постулаты *установимости* формулируются следующим образом [37, с. 49–50]:

1) конъюнкция $\mathcal{A} \wedge \mathcal{B}$ установима тогда и только тогда, когда установимо $\mathcal{A}$ и установимо $\mathcal{B}$;

2) дизъюнкция $\mathcal{A} \vee \mathcal{B}$ установима тогда и только тогда, когда установим хотя бы один из ее членов, т. е. когда установимо $\mathcal{A}$ или установимо $\mathcal{B}$;

3) импликация $\mathcal{A} \Rightarrow \mathcal{B}$ установима тогда и только тогда, когда установимость $\mathcal{A}$ может быть сведена к установимости $\mathcal{B}$, т. е. если всякий раз, когда будет установлено $\mathcal{A}$, мы сумеем из этого извлечь установимость $\mathcal{B}$;

4) константа $\perp$ (ложь) понимается как некоторое фиксированное высказывание, заведомо не являющееся установимым;

5) отрицание $\neg\mathcal{A}$ установимо тогда и только тогда, когда $\mathcal{A}$ неустановимо.

Клини (1945) предложил первый способ установления высказываний конструктивной логики, названный им *рекурсивной реализуемостью* (см. гл. 3).

2.4.4. Конструктивная теория множеств и топосы реализуемости

Классическая математика как теория описывается на языке теории множеств. Теория множеств – составляющая часть классической математической теории, и поскольку любая теория предполагает интерпретацию, то интерпретацией теории множеств является она сама. Теория множеств является топосом **Sets**. Поэтому можно сказать, что классическая теория можеств имеет в качестве своей интерпретации на языке теории категорий топос **Sets**.

В рамках конструктивной математики содержится теория, называемая *конструктивной теорией множеств.* Интерпретациями конструктивной теории множеств посредством топосов являются, в частности, *рекурсивный топос*, *топосы реализуемости* и *эффективный топос.*

2.5. Нормальные алгорифмы Маркова

Пусть дан алфавит A, т. е. некоторое непустое множество $A = \{a_1, a_2, ..., a_m\}$, из элементов которого мы составляем конечные последовательности вида $a_{i_1}...a_{i_k}$, называемые словами в алфавите A.

Формулы вида

$$\mathfrak{a} \to \mathfrak{b}, \quad \mathfrak{a} \to \cdot\ \mathfrak{b}, \tag{2.1}$$

где $\mathfrak{a}, \mathfrak{b}$ – слова в алфавите A, а символы $\to$ и $\cdot$ не входят в алфавит A, называются соответственно *простой* и *заключительной подстановками.*

Схема S состоит из конечного упорядоченного набора подстановок:

$$\mathfrak{a}_1 \to \delta_1\mathfrak{b}_1,\ ...,\ \mathfrak{a}_p \to \delta_p\mathfrak{b}_p,$$

где δ_j – либо пусто, либо есть символ $\cdot$.

Слово $\mathfrak{a}$ называется *вхождением* в слово $\mathfrak{m}$, если существуют слова $\mathfrak{r}_1$ и $\mathfrak{r}_2$ такие, что справедливо равенство: $\mathfrak{m} = \mathfrak{r}_1\mathfrak{a}\mathfrak{r}_2$.

Если имеется вхождение слова $\mathfrak{a}$ в слово $\mathfrak{m}$, т. е. $\mathfrak{m} = \mathfrak{r}_1\mathfrak{a}\mathfrak{r}_2$, то применение подстановки $\mathfrak{a} \to \delta\mathfrak{b}$ означает, что слово $\mathfrak{m}$ заменяется на слово $\mathfrak{r}_1\mathfrak{b}\mathfrak{r}_2$.

Нормальный алгорифм Маркова – это алфавит A, схема S и следующее правило переработки слов в алфавите A.

Пусть дано слово $\mathfrak{r}$. Полагаем по определению $\mathfrak{r}_0 = \mathfrak{r}$ и говорим, что слово $\mathfrak{r}_0$ получается из $\mathfrak{r}$ на шаге 0. Пусть сделано n шагов, в ходе которых получено слово $\mathfrak{r}_n$. Тогда делается $(n+1)$-й шаг. Сначала ищется вхождение для первого правила подстановки из схемы S. Если оно найдено, то происходит подстановка, и преобразуемое слово опять просматривается слева направо в поисках вхождения. Если вхождение для первого правила не найдено, то ищется вхождение для второго правила и т. д. Если вхождение найдено для j-й подстановки, то происходит подстановка, и просмотр правил начинается с первого, а слово просматривается сначала и слева направо. Возникшее в результате слово обозначается через $\mathfrak{r}_{n+1}$. Если была использована простая подстановка, то говорим, что слово $\mathfrak{r}_{n+1}$ получено из $\mathfrak{r}$ после $(n+1)$ шагов переработки. Если же была использована заключительная подстановка, то сразу после ее применения слово $\mathfrak{r}_{n+1}$ объявляется *результатом* применения алгорифма $\mathcal{M}$ и пишется

$$\mathfrak{r}_{n+1} = \mathcal{M}(\mathfrak{r}).$$

Может случиться, что ни одна из подстановок в схеме S не могла быть применена к слову $\mathfrak{r}_n$. Тогда пишем $\mathfrak{r}_{n+1} = \mathfrak{r}_n$ и говорим, что слово $\mathfrak{r}_{n+1}$ есть *результат* применения алгорифма $\mathcal{M}$. Если процесс переработки слова $\mathfrak{r}$ ни на каком шаге не обрывается, то говорим, что результат переработки слова $\mathfrak{r}$ с помощью алгорифма $\mathcal{M}$ *не определен*.

Пусть $f(\mathfrak{r})$ функция, определенная на словах алфавита A. Если для каждого слова $\mathfrak{r}$ в этом алфавите

$$f(\mathfrak{r}) = \mathcal{M}(\mathfrak{r}),$$

то говорим, что нормальный алгорифм Маркова $\mathcal{M}$ *вычисляет функцию* f.

Теорема 2.1. *Класс частичных функций, вычислимых на машине Тьюринга, совпадает с классом функций, вычислимых с помощью алгорифма Маркова.* ∎

Имеет место

> **Тезис Маркова.** *Все вычислимые функции вычисляются с помошью нормального алгорифма Маркова.*

2.6. Л.Э.Я. Брауэр

Брауэр (Brower) Люйтцен Эгбертус Ян (27 февраля 1881, Оверсхи, Нидерланды – 2 декабря 1966, Бларикум, Нидерланды) – голландский математик, логик, философ, основоположник интуиционизма и один из идейных предшественников и вдохновителей математического конструктивизма.

Окончил Амстердамский университет (1903), в котором позже защитил докторскую диссертацию (1907) и был старшим преподавателем с 1909 по 1951 год.


Л.Э.Я. Брауэр

В 1908 году в работе «О недостоверности логических принципов» изложил свои взгляды на построение математики и на ее основопологающие логические законы. Он заявил, что классическая логика является результатом неправомерного обобщения на бесконечные множества тех законов, которые были получены на небольших конечных множествах. Отказался принимать закон исключенного третьего на том основании, что для многих точно сформулированных утверждений не известна их истинность и ложность и нет даже мысленного процесса, который с гарантией приводил бы к решению данной проблемы. Предложил перестроить классическую математику

на основе другой, интуиционистской логики и другой интерпретации формул как задач на мысленные построения.

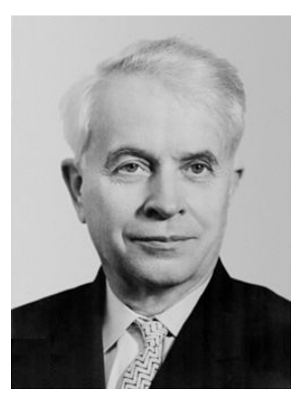
Брауэр говорил, что интуиция двуединства, основанная на разделении событий во времени на предшествующие и последующие, создает в мышлении человека натуральный ряд чисел. На основе понятия натурального ряда разум человека имеет возможность построения арифметики, а на ее фундаменте и всей математики.

Брауэр отрицательно отнесся к тезису Чёрча, считая, что алгоритмическая вычислимость не исчерпывает умственных построений.

В рамках классической математики Брауэром были получены, например, следующие выдающиеся математические результаты: теорема, доказывающая, что n-мерное евклидово пространство имеет топологическую размерность Лебега, равную n; теорема Брауэра о неподвижной точке; им было введено понятие гомотопической классификации отображений.

Погиб в 1966 г. в автокатастрофе.

2.7. А.А. Марков-младший



А.А.Марков-мл.

Андрей Андреевич Марков (9 [22] сентября 1903, Санкт-Петербург – 11 октября 1979, Москва) – советский математик, основоположник советской школы конструктивной математики. Сын великого русского математика А.А. Маркова.

По образованию А.А. Марков – физик. В 1924 г. он окончил физическое отделение физико-математического факультета Ленинградского университета, работал в Государственном физико-техническом институте. С 1925 по 1928 год был аспирантом,

а затем до 1935 года научным сотрудником Астрономического института. В 1933–1955 годах работал в Ленинградском университете (с 1936 года – профессор). С 1936 по 1942 и с 1943 по 1953 год – заведующий кафедрой геометрии. До июля 1942 года находился в блокадном Ленинграде.

Ученая степень доктора физико-математических наук была присвоена А.А. Маркову без защиты[2] диссертации в 1935 году. Член-корреспондент АН СССР (1953). Член КПСС с 1953 года.

С 1959 по 1979 год – заведующий кафедрой математической логики Московского государственного университета.

Одновременно с 1939 по 1972 год работал в Математическом институте им. В.А. Стеклова АН СССР. В 1950-х годах создал в Вычислительном центре АН СССР лабораторию математической логики и структуры машин, которой руководил около двадцати лет.

Наиболее известный результат – теорема об отсутствии алгоритма, классифицирующего односвязные 4-мерные гладкие многообразия. Автор понятия нормального алгорифма. Создал научную школу конструктивной математики и логики в СССР. Ученики: А.Г. Драгалин, Б.А. Кушнер, Н.М. Нагорный, Г.С. Цейтин, Н.А. Шанин и другие.

[2] В середине 1930-х годов в СССР были введены ученые степени кандидата и доктора наук. Ряду известных ученых степень доктора наук была присвоена без защиты диссертации. Из них сформировали советы, присваивающие ученые степени соискателям этих степеней.

Глава 3

Реализуемость Клини

В 1945 году Клини, используя уточненное понятие алгоритма, основанное на понятии рекурсивных функций, дал интерпретацию интуиционистской арифметики [25]. Идеи Клини существенным образом способствовали разработке основ конструктивной семантики.

Каждый конструктивный универсум представляет собой конечное или счетное множество объектов, допускающих эффективное описание в некотором языке [31, с. 44].

Конструктивная математика требует не просто констатации (доказательства) истинности некоторой формулы A, но и предоставления способа, конструкции, алгоритма, с помощью которого эта истинность устанавливается.

Так, с точки зрения интуициониста формула $\exists x \mathcal{A}(x)$ является неполным сообщением о том, как построить x, которое $\mathcal{A}(x)$. Более того, само $\mathcal{A}(x)$, в свою очередь, есть неполное сообщение о том, как установить, вычислить $\mathcal{A}(x)$.

Такие *неполные сообщения* должны быть восполнены путем задания общего алгоритма получения информации о том, как устанавливать истинность любых формул.

Алгоритмы – это, например, вычисление рекурсивных функций. Поэтому можно привлечь рекурсивные функции в качестве метода получения информации о том, как устанавли-

вать истинность формул. Клини использовал термин «рекурсивная реализация формул» для алгоритмического установления истинности формул. Вместо слова «истинная» он стал говорить о формуле, что она «рекурсивно реализуемая».

Рекурсивная реализуемость – понятие, лежащее в основе предложенного Клини метода конструктивного (интуиционистского) понимания математических и логических утверждений, который позволяет в точных терминах говорить об их «истинности» и является, следовательно, попыткой построения конструктивной семантики. Реализуемое утверждение – это утверждение, истинность которого проверяется с помощью конкретной вычислительной процедуры, т. е. с помощью конкретного алгоритма, который привязан к рассматриваемому утверждению.

Теория реализуемости должна показывать, как связываются математические и логические утверждения с алгоритмами, вычисляющими их истинность.

Предложенная Клини теория реализуемости – это попытка точного описания конструктивной истинности, которое не связано с привычным дедуктивным аппаратом интуиционистской (конструктивной) логики, она исходит непосредственно из самих принципов конструктивного понимания математических суждений [37, с. 278].

3.1. Гёделевская нумерация и индексы функций

3.1.1. Гёделевские номера

Сопоставим различные нечетные числа – номера – знакам формальной арифметики:

$\Rightarrow$	$\wedge$	$\vee$	$\neg$	$\forall$	$\exists$	$=$	$+$	$\cdot$	$\prime$	x_1	x_2	...
3	5	7	9	11	13	15	17	19	21	23	25	...

Поскольку любая формула ϕ формальной арифметики – это конечная запись (слово), состоящая из знаков, стоящих

в верхней строке таблицы, приведенной выше, то формуле ϕ соответствует последовательность из номеров использованных знаков:

$$a_0, a_1, a_2, a_3, a_4, ..., a_m.$$

Тогда формуле ϕ можно сопоставить число

$$g(\phi) = n = 2^{a_0} 3^{a_1} 5^{a_3} 7^{a_4} ... p_m^{a_m},$$

где $p_0 = 2, p_1 = 3, p_2 = 5$ и вообще p_m – m-е простое число.

Из этой записи видно, что в силу теоремы о единственности разложения любого натурального числа на простые множители, что каждому $n = g(\phi)$ однозначно соответствует набор $a_0, ..., a_m$ и, наоборот, каждому набору $a_0, ..., x_a$ однозначно соответствует число n.

Число $n = g(\phi)$ называется *гёделевским номером* рассматриваемой формулы.

Любое утверждение можно записать как формулу формальной арифметики, расширяющей логику предикатов. Гёделевская нумерация каждой формуле ставит в соответствие число, по которому формула легко восстанавливается. Поскольку любое (повествовательное) высказывание, касающееся арифметики, можно записать как формулу формальной арифметики, то формула несет некоторую *информацию*, и, следовательно, гёделевская нумерация ее кодирует в форме натурального числа.

Таким образом, вместо того, чтобы опрерировать со словесной информацией, можно иметь дело с натуральными числами, кодирующими словесную информацию.

Если дана последовательность формул

$$\phi_0, \phi_1, \phi_2, ..., \phi_m,$$

то гёделевский номер данной последовательности определяется как число

$$g(\phi_0, \phi_1, ..., \phi_m) = p_0^{g(\phi_1)} p_1^{g(\phi_2)} ... \ p_m^{g(\phi_m)}.$$

Таким образом, функция g взаимно однозначно отображает знаки, формулы и последовательности формул арифметики на подмножество натуральных чисел.

3.1.2. Представление частично рекурсивных функций. Индексы

Предикат $P(x_1, ..., x_n)$ называется *примитивно рекурсивным* (рекурсивным), если его характеристическая функция

$$\chi_P(x_1, ..., x_n) = \begin{cases} 1, & P(x_1, ..., x_n) = \top, \\ 0, & P(x_1, ..., x_n) = \bot \end{cases}$$

(или, что то же самое, если его множество истинности) примитивно рекурсивна (соответственно: рекурсивна).

Введем μ-оператор на предикатах:

$$\mu_y R(x_1, ..., x_n, y) = \min_{y \in \mathbb{N}} R(x_1, ..., x_n, y).$$

Клини показал, что существуют примитивно рекурсивный предикат $T(z, x_1, ..., x_n, y)$ и примитивно рекурсивная функция $U(z)$ такие, что для любой частично рекурсивной функции $f(x_1, ..., x_n)$ справедливо равенство

$$f(x_1, ..., x_n) = U(\mu_y T(e, x_1, ..., x_n, y)), \tag{3.1}$$

где e – гёделевский номер утверждений, с помощью которых проверяется вычислимость данной функции [36, с. 267].

Когда число z пробегает весь натуральный ряд $\mathbb{N}$, $U(\mu_y T(z, x_1, ..., x_n, y))$ дает некоторую нумерацию (с повторениями) всех частично рекурсивных функций от n переменных [36, с. 267].

Определение 3.1. *Индексом* частично рекурсивной функции $f : \mathbb{N}^n \to \mathbb{N}$ называется число $e \in \mathbb{N}$, для которого справедливо равенство (3.1).

Гёделевский номер утверждений, с помощью которых проверяется вычислимость данной функции f, является ее индексом. Каждая частично рекурсивная функция имеет бесконечно много индексов [36, с. 267].

Если $T(e, x_1, ..., x_n, y)$, то $U(y) = \{e\}(x_1, ..., x_n)$. Это понятно, если вспомнить, что e – номер (код, индекс) функции f, входящей в равенство (3.1). Иначе говоря, в формуле

$\{e\}(x_1, ..., x_n)$ код e функции $f(x_1, ..., x_n)$ заменяет ее собственный знак f.

3.2. Рекурсивная реализуемость

Алгоритмический подход к установлению истинности формулы состоит в том, чтобы связать формулу с информацией, которая позволяет эту истинность устанавливать. Формулы, а значит, информацию, которую они несут, кодируем гёделевскими номерами. Надо научиться связывать натуральные числа с формулами.

Клини вводит отношение между натуральными числами и формулами арифметики

$$(e \textbf{ реализует } \mathcal{A}),$$

которое должно выражать идею, что число e предоставляет, обеспечивает вычислительную (конструктивную) ясность относительно истинностного содержания формулы ϕ.

Интуитивно, «(e **реализует** $\mathcal{A}$)» имеет место в точности тогда, когда e есть полная информация, полный набор конструкций, нужный для обоснования $\mathcal{A}$ с алгоритмической, конструктивной точки зрения. Например, «(e **реализует** $(\mathcal{A} \Rightarrow \mathcal{B})$)» означает, что e есть алгоритм, перерабатывающий всякую реализацию $\mathcal{A}$ в реализацию $\mathcal{B}$ [16, с. 65].

Определение 3.2. (Клини, [25, с. 443–444]).

1) e реализует замкнутую элементарную[1] формулу P, если $e = 0$ и P истинна (другими словами, если $e = 0$ и P).

Пусть далее всюду $\mathcal{A}$ и $\mathcal{B}$ – замкнутые формулы. Тогда:

2) (e **реализует** $\mathcal{A} \wedge \mathcal{B}$), если $e = 2^a 3^b$, где (a **реализует** $\mathcal{A}$) и (b **реализует** $\mathcal{B}$);

3) (e **реализует** $\mathcal{A} \vee \mathcal{B}$), если $e = 2^0 3^a$, где (a реализует $\mathcal{A}$), или $e = 2^1 3^b$, где (b **реализует** $\mathcal{B}$);

[1] Формула называется элементарной, если она не содержит логических символов.

4) (e **реализует** ($\mathcal{A} \Rightarrow \mathcal{B}$)), если e есть гёделев номер частично рекурсивной функции f от одной переменной, такой, что для любого натурального числа a, если (a **реализует** $\mathcal{A}$), то $f(a)$ определено и ($f(a)$ **реализует** $\mathcal{B}$);

5) (e **реализует** $\neg\mathcal{A}$), если (e **реализует** $\mathcal{A} \Rightarrow (1 = 0)$);

6) (e **реализует** $\exists x \mathcal{A}(x)$), где x – переменная, а $\mathcal{A}(x)$ – формула, не содержащая никаких свободных переменных, кроме, быть может, x, если $e = 2^a 3^n$, где (n **реализует** $\mathcal{A}(a)$);

7) (e **реализует** $\forall x \mathcal{A}(x)$) (при тех же условиях на $\mathcal{A}(x)$ и x), если e есть гёделев номер общерекурсивной функции f от одной переменной, такой, что для каждого натурального n ($f(n)$ **реализует** $\mathcal{A}(n)$).

8) натуральное число e реализует элементарную замкнутую арифметическую формулу $t_1 = t_2$, если $e = 0$ и значение терма t_1 равно значению терма t_2, т. е. (e **реализует** ($t_1 = t_2$)) [25, с. 443].

Формула $\mathcal{A}$, не содержащая свободных переменных, *реализуема*, если существует число e, реализующее $\mathcal{A}$, т. е. (e **реализует** $\mathcal{A}$).

Формула $\mathcal{A}(y_1, ..., y_m)$, содержащая свободно только переменные $y_1, ..., y_m$ ($m > 0$), отличные друг от друга, называется (*рекурсивно*) *реализуемой*, если существует общерекурсивная функция f от m переменных такая, что для каждого набора натуральных чисел $n_1, ..., n_m$ число $f(n_1, ..., n_m)$ реализует $\mathcal{A}(n_1, ..., n_m)$.

Понятие (рекурсивной) реализуемости естественным образом распространяется и на формулы логики высказываний: пропозициональная формула называется реализуемой, если реализуема всякая арифметическая формула, получающаяся из данной путем подстановки.

Теорема 3.1 (Нельсон). *Любая доказуемая в конструктивном (интуиционистском) исчислении высказываний формула этого исчисления реализуема* [37, с. 293]. ■

Однако, как показал Дж. Роуз (1953), имеются реализуемые, но не выводимые формулы, т. е. конструктивное исчис-

ление высказываний неполно относительно понятия реализуемости [37, с. 306].

Отметим, «что всякая реализуемая пропозициональная формула выводима в классическом исчислении высказываний. В то же время В.П. Оревковым доказано существование реализуемых предикативных формул, невыводимых в классическом исчислении предикатов» [37, с. 308].

3.3. Формальная теория и ее классическая интерпретация

Формальная теория – это способ изложения логики без приписывания пропозициональным переменным, предикатам и формулам какого-либо значения.

3.3.1. Определение формальной теории

В основе формальной теории – *язык*, на котором «разговаривает теория». На первое место выходит *синтаксис*[2] этого языка, т. е. способ построения формул, в противопоставление *семантике*[3], озадаченной смыслом формул.

Формальная теория $\mathcal{T}$ состоит из следующих компонент:

1. Множества знаков, образующих *алфавит* языка теории.

2. Множества слов, составленных из знаков алфавита, называемых *формулами*.

3. Подмножества формул всего множества формул, называемых *аксиомами*.

---

[2] *Синтаксис* (от греч. syntaxis – «составление») – 1) характерные для конкретных языков средства и правила создания речевых единиц; 2) раздел грамматики, изучающий способы соединения слов в словосочетания и предложения, соединение предложений в сложные предложения [42].

[3] *Семантика* (от греч. semanticos – «обозначающий») – 1) смысловая сторона единиц языка, словосочетаний; 2) раздел логики, исследующий отношения логических знаков к понятиям и предметам действительности [42].

4. Множества *правил вывода*, с помощью которых из формул получают формулы.

В *язык* теории $\mathcal{T}$ входят алфавит и формулы. Количество аксиом может быть конечным или бесконечным.

Обычно формулы состоят из конечного числа знаков. Но бывают теории с бесконечно длинными формулами, т. е. с формулами, содержащими бесконечное число знаков.

Формальные теории построены для арифметики (формальная арифметика) (см.: [13]) и для классических логики высказываний и логики предикатов, а также для многих других математических содержательных теорий.

В каждой формальной теории определяется понятие доказательства формул.

Формальное доказательство формулы $\mathcal{A}$ в теории $\mathcal{T}$ – это конечная последовательность формул

$$\mathcal{A}_1, \mathcal{A}_2, ..., \mathcal{A}_n, \tag{3.2}$$

где каждая формула $\mathcal{A}_i$ является либо аксиомой, либо получена из предыдущих с помощью одного из правил вывода, а $\mathcal{A}_n$ – это формула $\mathcal{A}$.

Последняя формула в доказательстве (3.2) называется *теоремой.*

3.3.2. Интерпретация формальной теории

Смысл понятия интерпретации формальной теории состоит в приписывании *значения* каждой формуле теории. В результате о любой закрытой формуле $\mathcal{A}$ однозначно можно сказать, что формула либо истинная, т. е. принимает значение $\top$, либо ложная, т. е. $\mathcal{A} = \bot$.

Интерпретация состоит в указании конкретного непустого множества X, называемого областью интерпретации, и некоторого правила, по которому ставятся в соответстие:

– каждой переменной x – элемент $\bar{x} \in X$,

– каждому n-местному предикату $P(-, ..., -)$ – n-местное отношение $\bar{P} \subset X^n$ в X,

– функциям (операциям) n переменных $f(-, ..., -)$, участвующим в определении термов, – конкретные функции (операции) $\bar{f} : X^n \to X$ на X.

Формально *интерпретация* – это структура вида[4]

$$\mathcal{I} =< X; \bar{P}_1, ..., \bar{P}_p; \bar{f}_1, ..., \bar{f}_q > .$$

Содержанием при интерпретации формулы наполняются благодаря тому, что элементы множества X уже привязаны к конкретной реальности, знакомы и понятны. Например, в случае, когда $X = \mathbb{N}$, т. е. является множеством натуральных чисел, у нас не появляется чувства беспокойства, под формулами мы понимаем сведения из арифметики, которые прочно привязаны к практической деятельности инженеров, физиков, химиков и т. д. Следовательно, нам становится ясным, когда формула при определенной фиксации своих переменных *истинна*, когда – *ложна.*

Таким образом, при классической интерпретации формальной арифметики каждая замкнутая формула получает истинностное значение $\top$ (истина) или $\bot$ (ложь).

3.4. Формальная арифметика

Известная нам арифметика натуральных чисел $\mathbb{N}$ может быть изложена как формальная теория, называемая формальной арифметикой.

Теория, содержащая исчисление предикатов, называется *эгалитарной*, если она имеет дополнительный двуместный предикат $= (\cdot, \cdot)$, для которого выполняются две нелогические аксиомы:

1. $\forall x (= (x, x))$;
2. $= (x, y) \Rightarrow (\mathcal{A}(..., x, ..., y, ...) \Rightarrow \mathcal{A}(..., y, ..., y, ...))$.

Здесь $\mathcal{A}$ – произвольная формула.

---

[4] О понятии математической структуры см. в книге Н. Бурбаки «Теория множеств».

Вместо $=(x,y)$ пишут $x=y$. Таким образом, эгалитарная теория – это просто теория с *равенством.*

Определение 3.3. *Формальная арифметика* – это формальная теория с равенством, в которой дополнительно имеются:

1. Предметная константа 0.

2. Двуместные операции $+$ и $\cdot$ и одноместная операция $'$.

3. Знак равенства $\cdot = \cdot$.

4. Нелогические аксиомы равенства и следующие нелогические аксиомы арифметики:

$$
\begin{aligned}
&(\mathcal{A}(0) \wedge \forall x(\mathcal{A}(x) \Rightarrow \mathcal{A}(x'))) \Rightarrow \forall x \mathcal{A}(x),\\
&(t_1' = t_2') \Rightarrow (t_1 = t_2),\\
&\neg(t' = 0),\\
&(t_1 = t_2) \Rightarrow ((t_2 = t_3) \Rightarrow (t_1 = t_3)),\\
&(t_1 = t_2) \Rightarrow (t_1' = t_2'),\\
&t + 0 = t,\\
&(t_1 + t_2') = (t_1 + t_2)',\\
&t \cdot 0 = 0,\\
&t_1 \cdot t_2' \Rightarrow t_1 \cdot t_2 + t_1,
\end{aligned}
$$

где $\mathcal{A}$ – любая формула, t, t_1, t_2 – любые термы.

Первая аксиома – это известный способ доказательства посредством математической индукции. Если вместо t' писать $t+1$, то ясно, что t' – это следующее натуральное число, идущее за t. Другими словами, аксиомы формальной арифметики определяют натуральные числа

$$0, \quad 1 = 0', \quad 2 = 1', ..., \ \mathfrak{n}, \quad (n+1) = n', ...$$

и правила оперирования с ними с помощью операций сложения и умножения.

3.4.1. Интерпретации формальной арифметики

Множество натуральных чисел $\mathbb{N} = \{0, 1, 2, ..., n, ...\}$ с привычными нам операциями сложения и умножения является *стандартной интерпретацией* формальной арифметики. Существуют ли другие, *неизоморфные* интерпретации арифметики?

Определение 3.4. Интерпретация $\mathcal{I}$ *изоморфна* интерпретации J, если

1) $\mathcal{I}$ и $\mathcal{J}$ суть интерпретации одного и того же языка;

2) $\mathcal{I}$ и $\mathcal{J}$ приписывают одним и тем же пропозициональным символам одни и те же истинностные значения;

3) существует одно-однозначная функция h с областью определения, являющейся областью интерпретации $\mathcal{I}$, и множеством значений, совпадающим с областью интерпретации $\mathcal{J}$, такая, что

(a) если $\mathcal{I}$ приписывает имени значение d, то $\mathcal{J}$ приписывает этому имени значение $h(d)$;

(b) если $\mathcal{I}$ приписывает функциональному символу n-местную функцию f, то $\mathcal{J}$ приписывает этому символу функцию g такую, что для любых $d_1, ..., d_n, d$ из области $\mathcal{I}$ равенство $f(d_1, ..., d_n) = d$ имеет место тогда и только тогда, когда $g(h(d_1), ..., h(d_n)) = h(d)$;

(c) если $\mathcal{I}$ приписывает предикатному символу n-местную характеристическую функцию χ, то $\mathcal{J}$ приписывает ему характеристическую функцию $\widetilde{\chi}$ такую, что для любых $d_1, ..., d_n$ из области I выполняется $\chi(d_1, ..., d_n) = \widetilde{\chi}(h(d_1), ..., h(d_n))$.

Теорема 3.2. *Существует интерпретация*

$$\mathcal{I} =< N; \bar{0}, \bar{'}, \bar{+}, \bar{\cdot} >$$

формальной арифметики со счетным множеством N, неизоморфная арифметике $\mathbb{N}$ [5, с. 255]. ■

Интерпретации, неизоморфные $\mathbb{N}$, называются *нестандартными интерпретациями* формальной арифметики.

3.4.2. Рекурсивная интерпретация арифметики

В случае классической логики истинностных значений два – $\top$ (истина), $\bot$ (ложь), образующих двуэлементное множество истинностных значений $\Omega = \{\top, \bot\}$.

В случае интуиционистской логики истинностных значений может быть больше, и они могут рассматриваться как элементы некоторого множества Ω, содержащего и другие элементы, кроме $\top, \bot$.

Реализуемость Клини ассоциирует с каждым высказыванием $\mathcal{A}$ подмножество натуральных чисел $N_{\mathcal{A}}$, которые реализуют $\mathcal{A}$. Мы можем поэтому мыслить, что подмножество $N_{\mathcal{A}}$ – это истинностное значение формулы $\mathcal{A}$. Иначе говоря, в случае рекурсивной реализации мы принимаем, что множество Ω истинностных значений формул – это подмножество множества всех подмножеств $P(\mathbb{N})$ множества натуральных чисел $\mathbb{N}$, т. е. истинностные значения – это элементы множества $P(\mathbb{N})$.

Рекурсивная (*интуиционистская*) *интерпретация*

$$\mathcal{I} =< X, \bar{P}(\bar{x}), ..., \Omega >$$

строится следующим образом:

переменная $x \ \to \ \bar{x} \in X$,
предикат $P(x) \to \bar{P}(\bar{x})$, где $\bar{P} \in P(\mathbb{N})^X$, $\Omega \subset P(\mathbb{N})$,
предикат $\top \to \ \bar{\top}(\bar{x}) = \mathbb{N}$,
предикат $\bot \to \ \bar{\bot}(\bar{x}) = \emptyset$,
если $\mathcal{A}(x) \to \bar{\mathcal{A}}(\bar{x})$, $\mathcal{B}(x) \to \bar{\mathcal{B}}(\bar{x})$, то
$\overline{\mathcal{A}} \vee \overline{\mathcal{B}}, \ \bar{\mathcal{A}} \wedge \bar{\mathcal{B}}, \bar{\mathcal{A}} \Rightarrow \bar{\mathcal{B}} \in P(\mathbb{N})^X$,
задаваемые следующим образом:

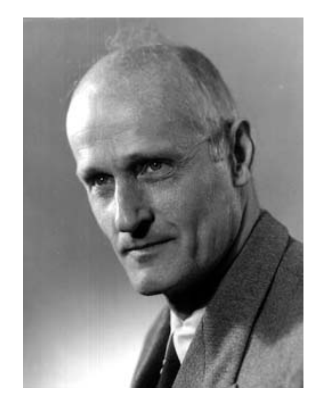
$$(\bar{\mathcal{A}} \wedge \bar{\mathcal{B}})(\bar{x}) = \{\langle n, m\rangle |\ n \in \bar{\mathcal{A}}(\bar{x}) \text{ и } m \in \bar{\mathcal{B}}(\bar{x})\},$$

$$(\bar{\mathcal{A}} \vee \bar{\mathcal{B}})(\bar{x}) = \{\langle 0, n\rangle |\ n \in \bar{\mathcal{A}}(\bar{x})\} \cup \{\langle 1, m\rangle |\ m \in \bar{\mathcal{B}}(\bar{x})\},$$

$$(\bar{\mathcal{A}} \Rightarrow \bar{\mathcal{B}})(\bar{x}) = \{e \in \mathbb{N}|\ \text{для всякого}\ n \in \bar{\mathcal{A}}(\bar{x}), \text{имеем}\ \{e\}(n) \in \bar{\mathcal{B}}(\bar{x})\},$$

где $\{e\}(n)$ – рекурсивная функция с гёделевским номером (кодом) e [107].

3.5. С.К. Клини


С.К. Клини

Стивен Коул Клини (Stephen Cole Kleene; 5 января 1909, Хартфорд, Коннектикут, США – 25 января 1994, Мадисон, Висконсин, США) – американский математик, ученик Чёрча.

Окончил Принстонский университет (1930). Получил докторскую ученую степень в 1934 году, защитив диссертацию «Theory of Positive Integers in Formal Logic».

Его работы совместно с работами Алонзо Чёрча, Курта Гёделя и Алана Тьюринга заложили основы теории вычислимости. Вел также исследования в области интуиционистской математики Брауэра и по теории конечных автоматов.

Во время Второй мировой войны служил в звании капитан-лейтенанта в Военно-морском флоте США (1942–1946). Был инструктором навигации в U.S. Naval Reserve's Midshipmen's School в Нью-Йорке, а затем директором проекта в Морской исследовательской лаборатории в Вашингтоне.

С 1946 года работал в Висконсинском университете (Мэдисон); с 1948 – профессор. Был научным руководителем по 13 докторским (Ph.D) диссертациям.

Среди логиков популярны его книги «Введение в метаматематику» (1952) и «Математическая логика» (1967). Целью первой книги он считал единое изложение математической логики и теории рекурсивных функций.

Предложенное Клини понятие рекурсивной реализуемости формул лежит в основе интуиционистской (конструктивной) интерпретации арифметических высказываний.

Глава 4

Категории и топосы

Топосы – это категории, «в сущности такие же», как теория множеств. Но их логика интуиционистская. Это делает их привлекательными для использования в конструктивистской математике.

4.1. Категории

Определение 4.1. *Категория* $\mathcal{K}$ – это:

1) объекты $A, B, C, ...$;

2) морфизмы $f, g, h, ...$;

3) каждый морфизм f связан с двумя объектами A, B; первый называют областью определения морфизма, а второй – областью значений. Используются обозначения $f : A \to B$ или $A \xrightarrow{f} B$;

4) для каждого объекта A имеется тождественный морфизм $1_A : A \to A$;

5) для каждой пары морфизмов $f : A \to B$ и $g : B \to C$ определена композиция морфизмов $g \circ f : A \to C$. Композиция должна удовлетворять двум условиям:

(i) Закон идентичности. Если дан морфизм $f : A \to B$, то $1_B \circ f = f$ и $f \circ 1_A = f$;

(ii) Закон ассоциативности. Если даны морфизмы $f : A \to B$, $g : B \to C$, $h : C \to D$, то $(h \circ g) \circ f = h \circ (g \circ f)$.

Совокупность морфизмов из A в B обозначают как $Hom_{\mathcal{K}}(A, B)$ или просто как $Hom(A, B)$.

Пример 4.1. Простым, но очень важным примером категории является теория множеств Кантора **Sets**. Объекты этой категории – это множества $A, B, C, D, ...$ а морфизмы – отображения $f : A \to B,\ g : C \to D, ...$ из множества в множество.

Пример 4.2. Теория топологических пространств **Top** также является примером категории. Ее объекты – это топологические пространства $X, Y, Z, T, ...$, а морфизмы – непрерывные отображения $f : X \to Y,\ g : Z \to T, ...$

Пример 4.3. Экзотическим примером категории является категория **N**. Она обладает только одним объектом, обозначаемым как N, а морфизмами являются натуральные числа $0, 1, ..., n, ...$, и имеют они запись в следующем виде: $0 : N \to N, 1 : N \to N, ..., n : N \to N, ...$ Композицией морфизмов n и m является натуральное число $n \circ m = n + m$. Единичная стрелка 1_N объекта N задается числом 0.

Двойственная категория. Если дана категория $\mathcal{K}$, то легко строится *двойственная категория* $\mathcal{K}^{op}$. Она имеет те же самые объекты, что категория $\mathcal{K}$, а морфизмы получаются из морфизмов категории $\mathcal{K}$ «обращением стрелки», т. е. если в $\mathcal{K}$ есть морфизм $f : A \to B$, то в $\mathcal{K}^{op}$ рассматривается морфизм $f : B \to A$.

Подкатегория. Категория $\mathcal{K}$ называется *подкатегорией* категории $\mathcal{E}$ (обозначается это через $\mathcal{K} \subseteq \mathcal{E}$), если

(i) каждый $\mathcal{K}$-объект является $\mathcal{E}$-объектом и

(j) для любых двух $\mathcal{K}$-объектов A и B морфизм $A \to B$ категории $\mathcal{K}$ есть морфизм категории $\mathcal{E}$.

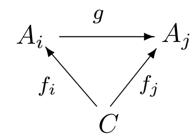
Подкатегория $\mathcal{K}$ называется *полной подкатегорией* категории $\mathcal{E}$, если для любых $\mathcal{K}$-объектов A и B выполняется равенство $Hom_{\mathcal{K}}(A, B) = Hom_{\mathcal{E}}(A, B)$, т. е. $\mathcal{E}$ не имеет морфизмов $A \to B$, не лежащих в $\mathcal{K}$.

4.2. Диаграммы

Определение 4.2. *Диаграммой* в категории $\mathcal{K}$ называется совокупность объектов $A_i, A_j, \ldots$ вместе с некоторыми морфизмами $f : A_i \longrightarrow A_j$ между отдельными объектами из этой диаграммы (между данной парой объектов может быть несколько морфизмов, а может и не быть их вовсе).

4.2.1. Конечно полные категории

Определение 4.3. *Конусом* для диаграммы $\mathcal{D}$ с объектами $A_i, A_j, \ldots$ называется такой объект C вместе с морфизмами $f_i : C \longrightarrow A_i$ для каждого объекта A_i из $\mathcal{D}$, что диаграмма



коммутативна для любого морфизма $g : A_i \longrightarrow A_j$ из $\mathcal{D}$, т. е. $g \circ f_i = f_j$.

Конус для диаграммы $\mathcal{D}$ обозначаем через $(f_i : C \longrightarrow A_i)$.

Определение 4.4. *Предел диаграммы* $\mathcal{D}$ есть конус $(f_i : C \longrightarrow A_i)$, такой, что для любого другого конуса $(f'_i : C' \longrightarrow A_i)$ для $\mathcal{D}$ существует единственный морфизм $f : C' \longrightarrow C$, для которого диаграмма

$$\begin{array}{ccc} & A_i & \\ {\scriptstyle f'_i}\nearrow & & \nwarrow{\scriptstyle f_i} \\ C' & \xrightarrow[f]{} & C \end{array}$$

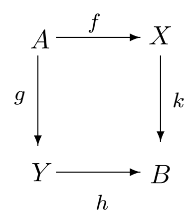
коммутативна для каждого объекта A_i из $\mathcal{D}$, т. е. $f'_i = f_i \circ f$.

Диаграмма *конечная*, если она состоит из конечного числа объектов и конечного числа морфизмов.

Определение 4.5. Категория $\mathcal{K}$ называется *конечно полной*, если она содержит предел любой конечной диаграммы.

4.2.2. Декартов квадрат. Обратный образ

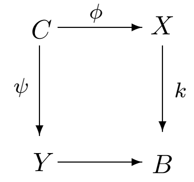
Диаграмма



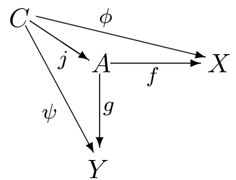
называется *декартовым квадратом*, если

1) она коммутативна, т. е. $k \circ f = h \circ g$;

2) для любых $\phi : C \longrightarrow X$, $\psi : C \longrightarrow Y$, для которых коммутативна диаграмма



существует единственный морфизм $j : C \longrightarrow A$ такой, что диаграмма

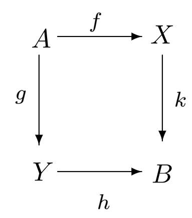


коммутативна.

Определение 4.6. *Обратный образ* (pullback) пары морфизмов $k : X \longrightarrow B$, $h : Y \longrightarrow B$ есть предел диаграммы ($k : X \longrightarrow B$, $h : Y \longrightarrow B$)

$$\begin{array}{ccc} & & X \\ & & \downarrow{\scriptstyle k} \\ Y & \xrightarrow[h]{} & B \end{array}$$

Другими словами, это объект A и морфизмы $f : A \longrightarrow X$, $g : A \longrightarrow Y$ такие, что квадрат



является декартовым.

Говорим, что в данном случае f *обратный образ* h *относительно* k, а g обратный образ k относительно h. Обозначение: $A = X \times_B Y$.

В категории **Sets** $X \times_B Y = \{(x, y) : k(x) = h(y)\} \subset X \times Y$.

Говорим о *категории с обратным образом*, если в ней обратный образ существует для любых пар морфизмов.

4.2.3. Морфизмы. Подобъекты

Морфизм $f : A \rightarrowtail B$ называется *мономорфизмом*, если для любой пары морфизмов $g : C \longrightarrow A$, $h : C \longrightarrow A$ из равенства $f \circ g = f \circ h$ следует $g = h$.

Для мономорфизма используется обозначение $f : A \rightarrowtail B$.

Подобъектом объекта D, или подобъектом в D, называется мономорфизм $f : A \rightarrowtail D$.

Морфизм $f : A \twoheadrightarrow B$ называется *эпиморфизмом*, если для любой пары морфизмов $g : B \longrightarrow A$, $h : B \longrightarrow A$ из равенства $g \circ f = h \circ f$ следует $g = h$.

Для эпиморфизма используется обозначение $f : A \twoheadrightarrow B$.

Теорема 4.1. (О факторизации морфизма). *Если дан морфизм $f : A \longrightarrow B$, то существует его эпи-моно-разложение* $A \overset{k}{\twoheadrightarrow} C \overset{h}{\rightarrowtail} B, \quad f = h \circ k.$ ■

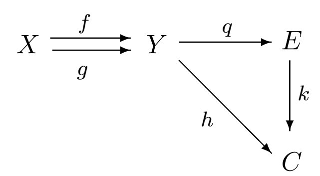
Произвольный морфизм $f : A \longrightarrow B$ называется *изоморфизмом*, если существует морфизм $g : B \longrightarrow A$ такой, что $f \circ g = 1_B$, $g \circ f = 1_A$.

Всякий изоморфизм является и мономорфизмом, и эпиморфизмом.

Коуравнитель пары морфизмов $f, g : X \longrightarrow Y$ есть морфизм $q : Y \longrightarrow E$ такой, что

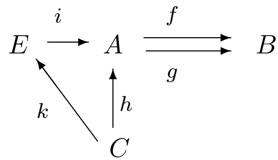
(i) $q \circ f = q \circ g$;

(ii) для любого морфизма $h : Y \longrightarrow C$, удовлетворяющего равенству $h \circ f = h \circ g$, существует единственная стрелка $k : E \longrightarrow C$, для которой диаграмма



коммутативна, т. е. $k \circ q = h$.

Уравнителем пары морфизмов $f, g : A \longrightarrow B$ называется морфизм $i : E \longrightarrow A$



такой, что

1) $f \circ i = g \circ i$;

2) для любого морфизма $h : C \longrightarrow A$ такому, что $f \circ h = g \circ h$, существует единственный морфизм $k : C \longrightarrow E$, для которого $i \circ k = h$.

4.2.4. Конечный и начальный объекты

Конечный объект 1. Пусть $\mathcal{D}$ – пустая диаграмма

т. е. без объектов и без морфизмов. Тогда конус для $\mathcal{D}$ – это просто любой объект (нет никаких морфизмов). Предел для

пустой $\mathcal{D}$ есть объект C такой, что для любого объекта C' существует единственный морфизм $f : C' \longrightarrow C$.

Этот удивительный объект C получает специальное название «*конечный объект*» и обозначается как **1**.

Объект **1** называется *конечным*, если для каждого объекта A существует один и только один морфизм из A в **1**.

В **Sets** конечный объект – это любое одноэлементное множество $\{*\}$.

Начальный объект 0. Объект **0** называется *начальным*, если для каждого объекта A существует один и только один морфизм из **0** в A, обозначаемый как $0_A : \mathbf{0} \longrightarrow A$.

В **Sets** начальный объект – это пустое множество $\emptyset$.

4.3. Функторы

Пусть даны две категории $\mathcal{K}$, $\mathcal{E}$. Если смотреть на категории как на множества, то хотелось бы иметь аналог отображения одной категории в другую. Таким аналогом является понятие функтора.

Функтором F из категории $\mathcal{K}$ в категорию $\mathcal{E}$ называется функция, которая ставит в соответствие:

1) каждому объекту A категории $\mathcal{K}$ – объект $F(A)$ категории $\mathcal{E}$;

2) каждому морфизму $f : A \longrightarrow B$ категории $\mathcal{K}$ – морфизм[1] $F(f) : F(A) \longrightarrow F(B)$ категории $\mathcal{E}$ такой, что

a) $F(1_A) = 1_{F(A)}$;

b) $F(g \circ f) = F(g) \circ F(f)$ для любых морфизмов f, g, для которых определена композиция $g \circ f$.

Тождественный функтор $\mathbf{1}_\mathcal{K} : \mathcal{K} \longrightarrow \mathcal{K}$ определяется правилами $\mathbf{1}_\mathcal{K}(A) = A, \mathbf{1}_\mathcal{K}(f) = f$. Эти же правила задают *функтор включения* $\mathcal{K} \hookrightarrow \mathcal{E}$ в случае, когда $\mathcal{K}$ – подкатегория в $\mathcal{E}$.

[1] Мы определяем *ковариантный* функтор; если направление морфизма меняется, т. е. $F(f) : F(B) \longrightarrow F(A)$, то имеем *контрвариантный* функтор.

Категории $\mathcal{C}, \mathcal{E}$ называются *эквивалентными*, если существуют функторы $F : \mathcal{K} \longrightarrow \mathcal{E}$, $G : \mathcal{E} \longrightarrow \mathcal{K}$, для которых $G \circ F = \mathbf{1}_{\mathcal{K}}$, $F \circ G = \mathbf{1}_{\mathcal{E}}$.

Функтор $F : \mathcal{K} \longrightarrow \mathcal{C}$ называется *точным* (faithful), если для любых $\mathcal{K}$-объектов K, K' и любых морфизмов $f, g : K \longrightarrow K'$, если $F(f) = F(g)$, то $f = g$.

Функтор $F : \mathcal{K} \longrightarrow \mathcal{C}$ является *полным* (full), если для любых $\mathcal{K}$-объектов K, K' и любого морфизма $\alpha : F(K) \longrightarrow F(K')$ существует $\mathcal{K}$-морфизм $f : K \longrightarrow K'$ такой, что $\alpha = F(f)$.

4.4. Категория функторов $\mathcal{E}^{\mathcal{K}}$

Пусть даны две категории $\mathcal{K}, \mathcal{E}$. Построим *категорию функторов* $\mathcal{E}^{\mathcal{K}}$, объектами которой являются функторы из $\mathcal{K}$ в $\mathcal{E}$.

Определим морфизмы категории $\mathcal{E}^{\mathcal{K}}$. Возьмем два функтора $F : \mathcal{K} \longrightarrow \mathcal{E}$ и $G : \mathcal{K} \longrightarrow \mathcal{E}$.

Морфизм $\tau : F \longrightarrow G$ *объекта* F *в объект* G называется *естественным преобразованием* функтора F в функтор G и состоит из семейства морфизмов $\{\tau_A : F(A) \longrightarrow G(A)$, где A – любой объект категории $\mathcal{K}\}$. Причём морфизм $\tau_A : F(A) \longrightarrow G(A)$ таков, что для любого морфизма $f : A \longrightarrow B$ диаграмма

$$\begin{array}{ccccc}
A & & F(A) & \xrightarrow{\tau_A} & G(A) \\
{\scriptstyle f}\big\downarrow & & {\scriptstyle F(f)}\big\downarrow & & \big\downarrow{\scriptstyle G(f)} \\
B & & F(B) & \xrightarrow{\tau_B} & G(B)
\end{array}$$

коммутативна.

Морфизмы τ_A называются *компонентами* преобразования τ.

4.5. Топосы

В теории множеств Кантора подмножество A однозначно характеризуется своей характеристической функцией

$$\chi_A(x) = \begin{cases} 1, & x \in A, \\ 0, & x \notin A. \end{cases}$$

Числа 0, 1 можно воспринять как истинностные значения $\perp, \top$. Поскольку теория множеств – это категория, то Ловер к концу 1960-х годов заметил, что двухэлементное множество $\{\perp, \top\}$ можно рассматривать как особый «объект истинностных значений» в категории множеств, который связан с классической двузначной логикой теории множеств. Ловер стал изучать категории, которые похожи на категорию множеств и обладают объектом, имеющим свойства, подобные объекту $\{\perp, \top\}$. В результате появилось понятие *классификатора подобъекта*, которое, в свою очередь, вводит в обиход понятие «подобъекта», являющегося категорным аналогом понятия подмножества. Классификатор подобъектов обозначается посредством Ω, и следствием существования такого объекта является всё то, что мы можем сказать о подобъектах A объекта D и что может быть переведено в разговор об отображениях D в Ω.

В итоге Ловер ввел понятие (элементарного) *топоса*, который был определён как декартово замкнутая категория с классификатором подобъектов.

Топосы оказались теми категориями, которые достаточно близки к категории множеств **Sets**. Теория множеств стала оплотом математики XX века, отчасти благодаря тому, что обладает возможностью легко порождать объекты, которые нашли широкое использование при решении самых разнообразных задач.

Речь идет о декартовых произведениях объектов, а также об объектах, состоящих из того, что на языке теории множеств называется множеством отображений из одного множества A в другое множество B, т. е. о множествах вида $B^A = \{f : A \to B\}$. Кроме того, учитывается то, что подмножество A множества D описывается с помощью характеристической функ-

ции χ_A, которая принимает значение в двухэлементном множестве $\{\bot, \top\}$. Эти свойства категории **Sets** берутся в качестве основных при сужении теории категорий до теории топосов.

4.5.1. Произведение объектов и морфизмов

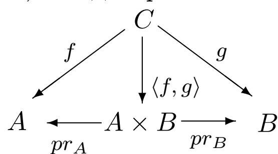
Определение 4.7. *Произведением* объектов A и B называется предел диаграммы, которая состоит только из двух объектов A и B и не имеет ни одного морфизма. Для произведения используется обозначение $A \times B$.

Поскольку $A \times B$ – конус, то имеем диаграмму

$$A \xleftarrow{pr_A} A \times B \xrightarrow{pr_B} B$$

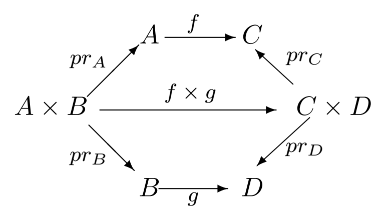
где морфизмы pr_A, pr_B называются *проекциями*.

Если дано произведение $A \times B$, то для всякого объекта C и морфизмов $f : C \longrightarrow A$, $g : C \longrightarrow B$ существует единственный морфизм $\langle f, g\rangle$ такой, что диаграмма



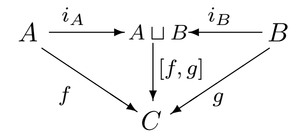
коммутативна.

Пусть даны объекты A, B, C, D и морфизмы $f : A \longrightarrow C$, $g : B \longrightarrow D$. Тогда существует единственный морфизм $f \times g : A \times B \longrightarrow C \times D$, для которого коммутативна диаграмма



Морфизм $f \times g : A \times B \longrightarrow C \times D$ называется *произведением морфизмов* f и g.

Определение 4.8. *Копроизведением* двух объектов A и B называется объект, обозначаемый через $A \sqcup B$, вместе с парой $(i_A : A \to A \sqcup B,\ i_B : B \to A \sqcup B)$ морфизмов такой, что для произвольной пары $(f : A \to C,\ g : B \to C)$ морфизмов существует один и только один морфизм $[f, g] : A \sqcup B \to C$, для которой диаграмма



коммутативна, т. е. $[f, g] \circ i_A = f$ и $[f, g] \circ i_B = g$. Морфизм $[f, g]$ называется *копроизведением* морфизмов f и g относительно *инъекций* i_A и i_B.

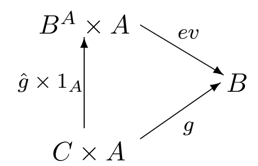
4.5.2. Экспоненциал

Экспоненциалы – это аналог возведения множества B в степень множества A. В теории множеств, напомним, $B^A = \{f : A \to B\}$, т. е. это множество, состоящее из всех отображений из A в B.

Определение 4.9. Категория $\mathcal{K}$ допускает *экспоненцирование*, если:

1) в ней существует произведение любых двух объектов;

2) для любых двух объектов A и B существует объект B^A («отображение» объекта A в объект B), называемый *экспоненциалом*, и морфизм $ev : B^A \times A \to B$, называемый *морфизмом значения*, такие, что для любых объекта C и морфизма $g : C \times A \to B$ существует единственный морфизм $\widehat{g} : C \to B^A$, для которых коммутативна диаграмма



Функция, ставящая в соответствие морфизму g морфизм $\hat{g}$, устанавливает биекцию

$$Hom_{\mathcal{K}}(C \times A, B) \cong Hom_{\mathcal{K}}(C, B^A)$$

между совокупностью $\mathcal{K}$-морфизмов из $C \times A$ в B и совокупностью $\mathcal{K}$-морфизмов из C в B^A.

Морфизмы g и $\hat{g}$, соответствующие друг другу на основании установленной биекции, называются *экспоненциально присоединенными* друг к другу.

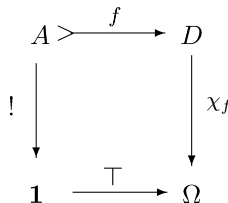
Многие категории обладают произведениями, но только некоторые имеют экспоненциалы.

Определение 4.10. Категории с произведениями, в которых каждая пара объектов имеет экспоненциал, называются *декартово замкнутыми* категориями.

4.5.3. Классификатор

Определение 4.11. *Классификатором подобъектов* для категории $\mathcal{K}$ называется объект Ω вместе с морфизмом $\top : \mathbf{1} \longrightarrow \Omega$ такой, что выполняется следующая

Ω-**аксиома**. Для каждого мономорфизма $f : A \rightarrowtail D$ существует единственный морфизм $\chi_f : D \longrightarrow \Omega$, для которого диаграмма



является декартовым квадратом.

Здесь через $! : A \longrightarrow \mathbf{1}$ обозначен единственный морфизм из A в $\mathbf{1}$.

Морфизм χ_f называется *характеристическим*, или *характером* морфизма f, а объект Ω – *классифицирующим объектом.* Морфизм $\top$ носит название «истина».

Термин «классификатор подобъектов» для Ω связан с тем, что любой мономорфизм $f : A \rightarrowtail D$ носит название *подобъекта* объекта D. Поэтому часто говорят, что морфизм $\chi_f : D \longrightarrow \Omega$ *классифицирует* подобъект $f : A \rightarrowtail D$.

В теории категорий подобъекты – это аналоги подмножеств $A \subset D$ в теории множеств.

Следствием существования классификатора подобъектов Ω является всё то, что мы можем сказать о подобъектах A объекта D и что может быть переведено в разговор об отображениях χ_f D в Ω [2].

4.5.4. Определение топоса

Определение 4.12. *Элементарный топос* – это декартово замкнутая категория, обладающая классификатором.

Элементарные топосы – это категории, которые очень близки по своим свойствам к категории теории множеств **Sets**. Действительно, если для топоса выполняется аксиома выбора, то он является булевым [15, с. 161].

Пример 4.4. Категория **Sets** является (булевым) топосом. Классификатором является множество $2 = \{0, 1\}$, а характеристические морфизмы – это характеристические функции $\chi_A : D \longrightarrow 2$, где $A \hookrightarrow D$ – подмножество множества D.

Пример 4.5. Категория $\mathbf{Sets}^{\mathcal{K}^{op}}$ – категория предпучков – является топосом, если $\mathcal{K}$ – малая категория, т. е. совокупность ее морфизмов образует множество.

[2] *Карпенко А.С.* Логика на рубеже тысячелетия.
URL: http://www.philosophy.ru/library/logic/karpenko/01.html.

Пример 4.6. Категория $Sh_J(\mathcal{K})$ – категория пучков над категорией $\mathcal{K}$, снабженной топологией Гротендика J, является топосом.

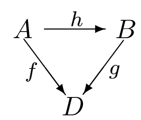
4.6. Логика топоса

Как известно, совокупность подмножеств некоторого множества образует булеву алгебру, с которой связана классическая двузначная логика. Поведение подобъектов в топосе характеризуется иной логикой, как правило, интуиционистской.

4.6.1. Алгебра подобъектов и ее логика

Дадим более точное определение подобъекта в топосе $\mathcal{K}$.

Пусть даны два мономорфизма $f : A \rightarrowtail D$ и $g : B \rightarrowtail D$. Они называются эквивалентными, $f \simeq g$, если существует изоморфизм $h : A \longrightarrow B$ такой, что коммутативна диаграмма



Данная эквивалентность является отношением эквивалентности на совокупности мономорфизмов в D. Класс эквивалентности с представителем f обозначаем через $[f]$ и вводим *семейство подобъектов*

$$Sub(D) = \{[f] \mid f : A \rightarrowtail D, A - \mathcal{K}\text{-объект}\}.$$

На $Sub(D)$ можно (см. ниже § 4.6.3 и [9]) определить операции $\cup, \cap, \setminus$ и отношение $\subset$, аналогичные операциям объединения, пересечения, дополнения и включения, определяемым в теории множеств, превратив ее в *алгебру подобъектов.*

Но, в отличие от теории множеств, $Sub(D)$, в общем случае, не является булевой алгеброй. Поскольку булева алгебра связана с классической двузначной логикой, то говорят, что

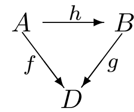
топосы, в общем случае, подчиняются неклассической логике.

Топос называется *булевым*, если $Sub(D)$ является булевой алгеброй. Для булева топоса справедлив закон исключённого третьего [9, с. 172] и любые другие утверждения, которые доказуемы в рамках классической логики [9, с. 173].

Таким образом, в произвольном топосе, в общем случае, действуют законы интуиционистской логики, а совокупность $Sub(D)$ является так называемой *алгеброй Гейтинга* [9, с. 196].

4.6.2. Категория $Sub(D)$

Объекты категории $Sub(D)$ – это подобъекты $A \rightarrowtail D$. Если $f : A \rightarrowtail D$ и $g : B \rightarrowtail D$ два подобъекта, то $Sub(D)$-морфизм из A в B, задающий еще и отношение порядка $A \subseteq B$, – это морфизм $h : A \longrightarrow B$ такой, что диаграмма



коммутативна.

Предложение 4.1. *$Sub(D)$ обладает структурой частично упорядоченного множества и является категорией, точнее примером категории порядка.* ■

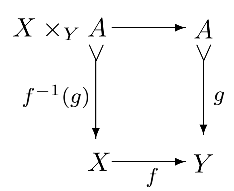
4.6.3. Функтор обратного образа f^{-1}

Рассмотрим морфизм $f : X \longrightarrow Y$. Тогда существует функтор, который записываем как

$$f^{-1} : Sub(Y) \longrightarrow Sub(X)$$

и называем *функтором обратного образа.*

В самом деле, если дан мономорфизм $g : A \rightarrowtail Y$, то существует обратный образ



который дает мономорфизм $f^{-1}(g) : X \times_Y A \rightarrowtail X$. Можно показать, что f^{-1} корректно определен на классах эквивалентности $[g]$, т. е. если g' представляет класс $[g]$, то $f^{-1}(g') \in [f^{-1}(g)]$. Тем самым f^{-1} является морфизмом алгебр Гейтинга $Sub(Y)$ и $Sub(X)$.

4.6.4. Логические связки в топосе

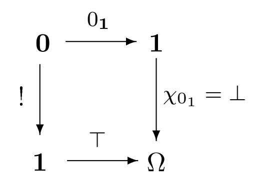
В **Sets** логические связки $\neg, \wedge, \vee, \Rightarrow$ являются отображениями множества истинности $\{\bot, \top\}^k$ в $\{\bot, \top\}$ $(k = 1, 2)$.

Поскольку классификатор Ω в топосе играет роль множества истинности $\{\bot, \top\}$ в **Sets**, то можно определить логические связки $\neg, \cap, \cup, \Rightarrow$, являющиеся аналогами логических связок $\neg, \wedge, \vee, \Rightarrow$ в **Sets**, как морфизмы Ω^k в Ω.

Пусть $\mathcal{K}$ – некоторый топос с классификатором подобъектов $\top : \mathbf{1} \longrightarrow \Omega$.

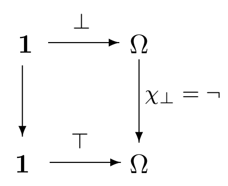
«Истина». Это морфизм $\top : \mathbf{1} \longrightarrow \Omega$.

«Ложь». Морфизм $\bot : \mathbf{1} \longrightarrow \Omega$ является характером для (единственного) морфизма $0_{\mathbf{1}} : \mathbf{0} \longrightarrow \mathbf{1}$, т. е. квадрат



является декартовым.

Логическое отрицание. Морфизм $\neg : \Omega \longrightarrow \Omega$ есть единственный морфизм, для которого квадрат



декартов. Следовательно, $\neg$ – это характер $\chi_\top$.

Конъюнкция. Морфизм $\cap : \Omega \times \Omega \longrightarrow \Omega$ есть характер морфизма $\langle \top, \top \rangle : \mathbf{1} \longrightarrow \Omega \times \Omega$, т. е. имеем декартов квадрат

$$\begin{array}{ccc} \mathbf{1} & \xrightarrow{\langle \top, \top \rangle} & \Omega \times \Omega \\ {\scriptstyle !}\downarrow & & \downarrow {\scriptstyle \chi_{\langle \top, \top \rangle} = \cap} \\ \mathbf{1} & \xrightarrow{\ \ \top\ \ } & \Omega \end{array}$$

Дизъюнкцию $\cup : \Omega \times \Omega \longrightarrow \Omega$ и **импликацию** $\Rightarrow : \Omega \times \Omega \longrightarrow \Omega$ см. в [9, с. 152].

4.7. Интерпретации в топосах формальных языков 1-го порядка

Интерпретацией для формальной теории (языка 1-го порядка) $\mathbf{L}$ *в топосе* $\mathcal{E}$ называется структура

$$\mathcal{I} =< A, r, f_c >,$$

в которой

1) A – произвольный непустой объект, т. е. $Hom_{\mathcal{E}}(\mathbf{1}, A) \neq \emptyset$;
2) $r : A \times A \longrightarrow \Omega$ – произвольная стрелка;
3) $f_c : \mathbf{1} \longrightarrow A$ – произвольный «элемент» объекта A.

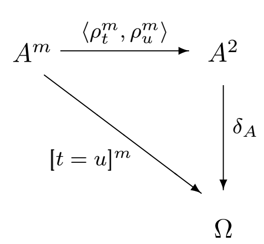
Для каждого терма t и подходящего m определим морфизм ρ_t^m:

$$\rho_t^m = \begin{cases} pr_i : A^m \longrightarrow A, & \text{если } t = v_i, \\ f_c \circ\, ! : A^m \longrightarrow A, & \text{если } t = c. \end{cases}$$

Здесь $pr_i^m : A^m \longrightarrow A$ – морфизм-проекция, участвующий в определении произведения A^m; v_i – индивидуальная переменная теории **L**; c – константа теории **L** (примем, что она единственная).

Для каждой формулы ϕ теории **L** и каждого подходящего m индуктивно определим морфизм $[\![\phi]\!]^m : A^m \longrightarrow \Omega$, являющийся *интерпретацией этой формулы* в $\mathcal{I}$, следующим образом:

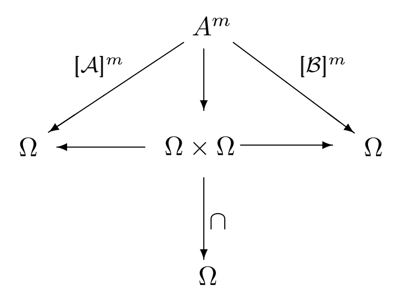
1) (интерпретация равенства) $[\![t = u]\!]^m = \delta_A \circ \langle \rho_t^m, \rho_u^m \rangle$:



где $\delta_A : A \times A \longrightarrow \Omega$ – характер для Δ_A, а $\Delta_A : A \rightarrowtail A \times A$ – мономорфизм $\langle 1_A, 1_A \rangle$;

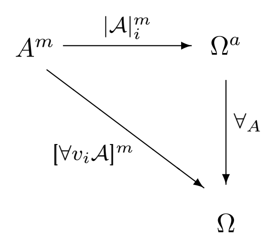
2) (интерпретация предиката $R(\cdot,\cdot)$) $[\![R(t,u)]\!]^m = r \circ \langle \rho_t^m, \rho_u^m \rangle$;

3) (интерпретация конъюнкции) $[\![\mathcal{A} \wedge \mathcal{B}]\!]^m = [\![\mathcal{A}]\!]^m \cap [\![\mathcal{B}]\!]^m = \cap \circ \langle [\![\mathcal{A}]\!]^m, [\![\mathcal{B}]\!]^m \rangle$,



где $\cap : \Omega \times \Omega \longrightarrow \Omega$ – морфизм, задающий конъюнкцию в топосе;

4) (интерпретация дизъюнкции) $[\![\mathcal{A} \vee \mathcal{B}]\!]^m = [\![\mathcal{A}]\!]^m \vee [\![\mathcal{B}]\!]^m$;
5) (интерпретация отрицания) $[\![\neg\mathcal{A}]\!]^m = \neg \circ [\![\mathcal{A}]\!]^m$;
6) (интерпретация импликации) $[\![\mathcal{A} \Rightarrow \mathcal{B}]\!]^m = [\![\mathcal{A}]\!]^m \Rightarrow [\![\mathcal{B}]\!]^m$;
7) (интерпретация $\forall v_i \mathcal{A}$) $[\![\forall v_i \mathcal{A}]\!]^m = \forall_A \circ |\mathcal{A}|_i^m$,



где

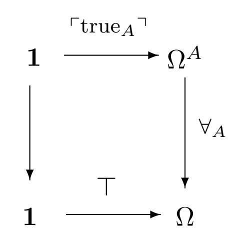
$|\mathcal{A}|_i^m$ – морфизм, экспоненциально присоединенный к композиции

$$A^{m+1} \overset{T_i^{m+1}}{\longrightarrow} A^m \overset{[\![\mathcal{A}]\!]^m}{\longrightarrow} \Omega,$$

$T_i^{m+1} : A^{m+1} \longrightarrow A^m$ $(1 \leq i \leq m)$ – морфизм

$$\langle pr_1^{m+1}, ..., pr_{i-1}^{m+1}, pr_{m+1}^{m+1}, pr_{i+1}^{m+1}, ..., pr_m^{m+1} \rangle;$$

$\forall_A : \Omega^A \longrightarrow \Omega$ – единственный морфизм, делающий диаграмму



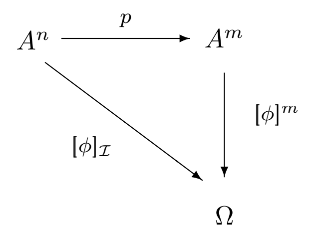
декартовым квадратом, $\ulcorner \mathrm{true}_A \urcorner$ – морфизм, экспоненциально присоединенный к композиции $\mathrm{true}_A \circ pr_A : \mathbf{1} \times A \longrightarrow A \longrightarrow \Omega$, $\mathrm{true}_A = \top \circ \, ! : A \overset{!}{\longrightarrow} \mathbf{1} \overset{\top}{\longrightarrow} \Omega$ (обозначение из [9, с. 96]);

8) (интерпретация $\exists v_i \mathcal{A}$) $[\![\exists v_i \mathcal{A}]\!]^m = \exists_A \circ |\mathcal{A}|_i^m$ (см.: [9, с. 258]).

Пусть формула $\phi(v_1, ..., v_{i_n})$ имеет n свободных переменных и g – произвольный морфизм из A^n в A. Выберем какое-нибудь подходящее для ϕ число m. Обозначим через $p = \langle p_1, ..., p_m \rangle$, $p : A^n \longrightarrow A^m$, где

$$p_j = \begin{cases} pr_k^n : A^n \longrightarrow A, & \text{если } \ j = i_k \ \text{ для некоторого } \ k \leq n, \\ g, & \text{в противном случае.} \end{cases}$$

Определим морфизм $[\![\phi]\!]_{\mathcal{I}} = [\![\phi]\!]^m \circ p$,



Определение 4.13. Говорим, что $\mathcal{I}$ есть *интерпретация в топосе $\mathcal{E}$ для формулы ϕ*, или символически

$$\mathcal{I} \models_{\mathcal{E}} \phi,$$

если $[\![\phi]\!]_{\mathcal{I}} = \mathrm{true}_{A^n}$, где $\mathrm{true}_{A^n} : A^n \overset{!}{\longrightarrow} \mathbf{1} \overset{\top}{\longrightarrow} \Omega$, $\mathrm{true}_{A^n} = \top \circ \,!$.

Заметим, что если $n \geq 1$, то мы могли бы взять в качестве g произвольную проекцию $A^n \longrightarrow A$, в то время как при $n = 0$ надо предполагать, что A непуст, чтобы существовала стрелка $g : 1 \longrightarrow A$.

Определение 4.14. Формула ϕ называется *$\mathcal{E}$-общезначимой*, $\mathcal{E} \models \phi$, если для любой интерпретации $\mathcal{I}$ имеет место $\mathcal{I} \models_{\mathcal{E}} \phi$.

Теорема 4.2. *Если $\vdash_{IL} \phi$, то $\mathcal{E} \models \phi$.* ■

Таким образом, доказуемые формулы интуиционистской логики являются общезначимыми в топосе.

4.8. Вложение Ионеды

Вложение Ионеды (Yoneda) – это функтор

$$y : \mathcal{K} \longrightarrow \mathbf{Sets}^{\mathcal{K}^{op}}$$

такой, что

(1) каждому $A \in \mathcal{K}$ соответствует функтор

$$y(A) : \mathcal{K}^{op} \longrightarrow \mathbf{Sets},$$

$$y(A)(-) = Hom_{\mathcal{K}}(-, A) \in \mathbf{Sets};$$

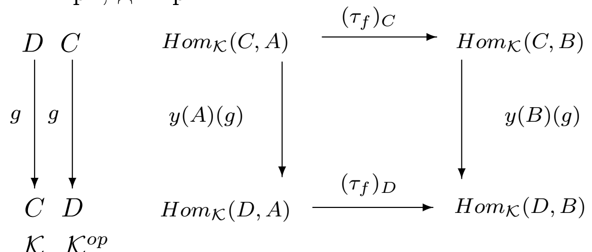
(2) каждому морфизму $f \in Hom_{\mathcal{K}}(A, B)$ соответствует естественное преобразование τ_f функтора $y(A)$ в функтор $y(B)$ с компонентами:

$$(\tau_f)_{(-)} : Hom_{\mathcal{K}}(-, A) \longrightarrow Hom_{\mathcal{K}}(-, B),$$

$$\{(\tau_f)_C : C \in \mathcal{K}\} \equiv$$

$$\equiv \{(\tau_f)_C : Hom_{\mathcal{K}}(C, A) \longrightarrow Hom_{\mathcal{K}}(C, B)\}.$$

Иначе говоря, диаграмма



коммутативна, где $y(A)(g) : y(A)(C) \longrightarrow y(A)(D)$ – морфизм, который сопоставляется морфизму $g : C \longrightarrow D$ для функтора $y(A)$ в соответствии с определением понятия функтора (см. § 4.4).

Иначе говория, если $h : C \longrightarrow A$ и $g : D \longrightarrow C$ – морфизм в $\mathcal{K}$, то $g : D \longrightarrow C$ морфизм в $\mathcal{K}$. Следовательно, $h \circ g : D \longrightarrow A$ – морфизм в $\mathcal{K}$ и поэтому

$$y(A)(g) : h \longrightarrow h \circ g,$$

$$y(A)(g) : Hom_{\mathcal{K}}(C, A) \longrightarrow Hom_{\mathcal{K}}(D, A).$$

4.9. Топос пучков

4.9.1. Топология Гротендика

Определение 4.15. *Топологией Гротендика*[3] J на категории $\mathcal{K}$ называется функция, сопоставляющая каждому объекту A совокупность $J(A)$ множеств морфизмов, оканчивающихся в A, такая, что

(i) одноэлементное множество $\{1_A : A \longrightarrow A\}$ принадлежит $J(A)$;

(ii) если $\{f_x : A_x \longrightarrow A \mid x \in X\} \in J(A)$ и для каждого $x \in X$

$$\{f_y^x : A_y^x \longrightarrow A_x \mid y \in X_x\} \in J(A_x),$$

то

$$\{f_x \circ f_y^x : A_y^x \longrightarrow A \mid x \in X \text{ и } y \in X_x\} \in J(A);$$

(ij) если $\{f_x : A_x \longrightarrow A \mid x \in X\} \in J(A)$ и $g : B \longrightarrow A$ – произвольный морфизм, то для каждого $x \in X$ существует обратный образ морфизма f_x относительно g

$$\begin{array}{ccc} B \times_A A_x & \longrightarrow & A_x \\ {\scriptstyle g_x}\downarrow & & \downarrow{\scriptstyle f_x} \\ B & \xrightarrow[g]{} & A \end{array}$$

и

$$\{g_x : B \times_A A_x \longrightarrow B \mid x \in X\} \in J(B).$$

Определение 4.16. Категория $\mathcal{K}$ с топологией J, т. е. пара $(\mathcal{K}, J)$, называется *ситусом* (site), а элементы из J – *покрытиями.*

Если элементы $\{f_x : A_x \longrightarrow A \mid x \in X\}$ топологии J таковы, что копроизведения $\coprod_{x \in X} A_x$ существуют и $[f_x]_{x \in X} : \coprod_{x \in X} A_x \longrightarrow A$ есть изоморфизм, то топология Гротендика на $\mathcal{K}$ называется *топологией копроизведения.*

[3] В книге Гольдблата [9, с. 385] это названо *предтопологией.*

Замечание 4.1. Топология Гротендика – это структура на категории, которая делает ее объекты похожими на открытые множества топологического пространства.

Замечание 4.2. Топология Гротендика говорит, каким образом объекты ситуса $\mathcal{C}$ могут быть «покрыты» посредством морфизмов другими объектами. В случае, когда I – топологическое пространство и рассматриваемые объекты – это открытые множества в I, то топология Гротендика говорит, каким образом открытые множества A покрываются в смысле операции включения $A_x \hookrightarrow A$ семейством открытых множеств $J(A) = \{A_x\}_{x\in X}$ таким, что

$$\bigcup_{x\in X} A_x = A.$$

Таким образом, топология Гротендика – это структура на категории $\mathcal{K}$, которая делает её объекты похожими на открытые множества топологического пространства I, образующие категорию $\mathcal{O}(I)$. Топологии Гротендика аксиоматизируют определение открытого покрытия.

Существует естественный способ сопоставить топологическому пространству I топологию Гротендика $(\mathcal{O}(I), J)$, поэтому топологии Гротендика часто рассматривают как обобщение обычных топологий. Объекты категории $\mathcal{O}(I)$ – открытые множества топологии пространства I, морфизмы между двумя объектами V, U – это включения $V \hookrightarrow U$, а покрытиями $J(U)$ являются так называемые *решёта.*

«Для большого класса топологических пространств действительно можно восстановить топологию пространства I по её топологии Гротендика, однако уже для антидискретного пространства это не так»[4].

4.9.2. Категории предпучков и пучков

Пусть $\mathcal{K}$ – категория.

Предпучок – это функтор $F : \mathcal{K}^{op} \longrightarrow \mathbf{Sets}$. Категория $\mathbf{Sets}^{\mathcal{K}^{op}}$, состоящая из таких функторов и естественных преобразований, называется *категорией предпучков.*

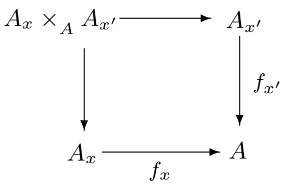
[4] См. статью «Топология Гротендика» в Википедии.

Категорию предпучков часто обозначают как $St(\mathcal{K})$, т. е.

$$St(\mathcal{K}) = \mathbf{Sets}^{\mathcal{K}^{\mathbf{op}}}.$$

Она является топосом.

Пусть J – топология Гротендика на категории $\mathcal{K}$ и $\{f_x : A_x \longrightarrow A|\ x \in X\} \in J(A)$. Для каждых $x, x' \in X$ построим декартов квадрат



Пусть F – предпучок над $\mathcal{K}$. Он определяет функции $F_{xx'} : F(A_x) \longrightarrow F(A_x \times_A A_{x'})$ и $F_{x'x} : F(A_{x'}) \longrightarrow F(A_x \times_A A_{x'})$, являющиеся F-образами двух новых морфизмов $f_x, f_{x'}$, получаемых при построении декартова квадрата. Обозначим также через F_x стрелку $F(f_x) : F(A) \longrightarrow F(A_x)$.

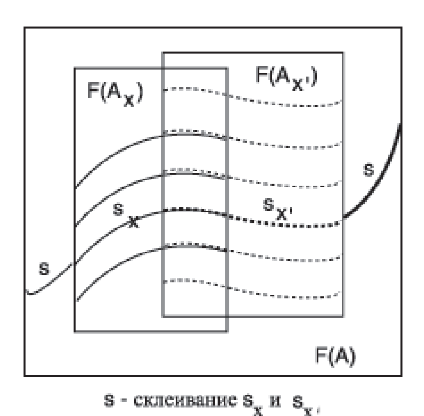


Рис. 4.1. Пучок – это предпучок с семейством «сечений» $\{s_x : x \in X\}$, склеенным в единое «сечение» s

Определение 4.17. Предпучок F называется *пучком над ситусом* $(\mathcal{K}, J)$, если он удовлетворяет следующему условию:

COM. *Пусть* $\{f_x : A_x \longrightarrow A | \ x \in X\} \in J(A)$ *– произвольное покрытие объекта* A *и* $\{s_x : x \in X\}$ *– произвольное семейство элементов* $s_x \in F(A_x)$, *являющихся попарно совместимыми, т. е. для любых* $x, x' \in X$ *выполняется равенство* $F_{xx'}(s_x) = F_{x'x}(s_{x'})$. *Тогда существует единственный* $s \in F(A)$, *удовлетворяющий равенству* $F_x(s) = s_x$ *при любом* $x \in X$.

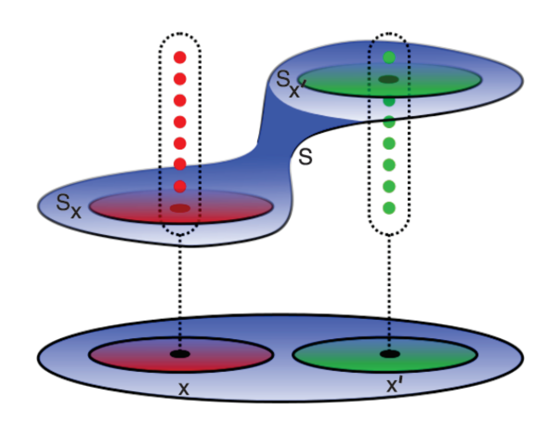


Рис. 4.2. Склеивание в пучке [96]

Элемент $s \in F(A)$ часто называют *склеивающим* (gluing) семейство $\{s_x : x \in X\}$ (см. рис. 4.1, 4.2) [58, p. 89].

Определение 4.18. Полную подкатегорию категории предпучков $St(\mathcal{K})$, образованную пучками над ситусом $(\mathcal{K}, J)$, обозначаем через $Sh_J(\mathcal{K})$ (или $Sh(\mathcal{K}, J)$) и называем *категорией пучков.*

Таким образом,

$$Sh_J(\mathcal{K}) \subset \mathbf{Sets}^{\mathcal{K}^{op}}.$$

Предпучок $F : \mathcal{K}^{op} \longrightarrow \mathbf{Sets}$, для которого справедливо равенство $F(-) = y(A)(-)$ для некоторого объекта $A \in \mathcal{K}$, где y – вложение Ионеды, называется *представленным.*

Топология J на категории $\mathcal{K}$ называется *подканонической*, если

все представленные предпучки на $\mathcal{K}$ являются пучками. *Каноническая топология* – это наибольшая подканоническая топология на $\mathcal{K}$.

Для подканонической (канонической) топологии

$$y(\mathcal{K}) \hookrightarrow Sh_J(\mathcal{K}).$$

Теорема 4.3. *Для каждого класса предпучков существует единственная наибольшая топология, в которой они являются пучками* [15, с. 35]. ■

4.9.3. Топосы Гротендика

Определение 4.19. *Топосом Гротендика* называется категория, эквивалентная категории вида $Sh_J(\mathcal{K})$.

Не все топосы являются топосами Гротендика. Топосы реализуемости над частичными комбинаторными алгебрами (pca), которые описываются в гл. 6, не являются таковыми.

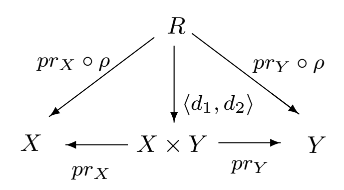
4.10. Пополнение категорий

4.10.1. Отношения эквивалентности на объектах. Фактор-объекты

Пусть дана категория $\mathcal{K}$ с обратными образами.

Отношение между ее объектами X и Y – это мономорфизм $\rho : R \rightarrowtail X \times Y$.

Для отношения ρ диаграмма



коммутативна. Следовательно, как следует из определения произведения объектов, $\rho = \langle d_1, d_2 \rangle$, где $d_1 = pr_X \circ \rho$, $d_2 = pr_Y \circ \rho$.

Если даны два отношения $\rho, \rho' : R \rightarrowtail X \times Y$, то пишем $\rho \subseteq \rho'$, если $\rho = \rho' \circ \sigma$ для некоторого $\sigma : R \longrightarrow R'$.

Диагональное отношение – это мономорфизм $\delta_X : X \rightarrowtail X \times X$.

Если дано отношение $\rho : R \rightarrowtail X \times Y$, обозначаем через ρ° отношение $R \rightarrowtail Y \times X$.

Определение 4.20. Даны отношения $\rho : R \rightarrowtail X \times Y$ и $\sigma : S \rightarrowtail Y \times Z$. Определяем их *композиционное отношение* $\sigma.\rho$ следующим образом [82, p. 45]. Рассматриваем декартов квадрат (обратный образ)

$$\begin{array}{ccc} P & \xrightarrow{k} & S \\ {\scriptstyle h}\downarrow & & \downarrow{\scriptstyle pr_1 \circ \sigma} \\ R & \xrightarrow[pr_2 \circ \rho]{} & Y \end{array}$$

и полагаем, что $\sigma.\rho : T \rightarrowtail X \times Z$ есть мономорфизм[5], появляющийся при факторизации морфизма $\langle pr_1 \circ \rho \circ h, pr_2 \circ \sigma \circ k\rangle : P \longrightarrow X \times Z$:

$$P \twoheadrightarrow T \rightarrowtail X \times Z$$

(см. теорему 4.1, [9, с. 127]).

В любой категории с обратным образом *отношение эквивалентности* на объекте X есть отношение $\rho : R \rightarrowtail X \times X$ такое, что [82, p. 46]

$$\begin{aligned} &\delta_X \subseteq \rho \quad \text{(рефлексивность)}, \\ &\rho^\circ \subseteq \rho \quad \text{(симметричность)}, \\ &\rho.\rho \subseteq \rho \quad \text{(транзитивность)}. \end{aligned} \tag{4.1}$$

Фактор-объект (quotient) X/R объекта X в $\mathcal{K}$ по отношению эквивалентности $\rho = \langle d_1, d_2\rangle : R \rightarrowtail X \times X$ – это такой объект в $\mathcal{K}$, который дается диаграммой коуравнителя

$$R \underset{d_2}{\overset{d_1}{\rightrightarrows}} X \xrightarrow{\quad q \quad} X/R$$

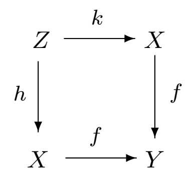
в предположении, что такой коуравнитель существует. Если морфизм существует, то он всегда является эпиморфизмом.

О фактор-объекте X/R говорят, что он получен при факторизации объекта X по отношению эквивалентности R.

[5] Мономорфизм обозначают как $\rightarrowtail$, а эпиморфизм как $\twoheadrightarrow$.

4.10.2. Регулярная категория

Пусть дана категория $\mathcal{K}$ и морфизм $f : X \longrightarrow Y$. Два морфизма $h, k : Z \longrightarrow X$ образуют *ядерную пару* для f, если диаграмма



есть декартов квадрат (pullback).

Определение 4.21. Категория называется *регулярной*, если она имеет конечные пределы, коуравнители ядерных пар и обратный образ регулярного эпиморфизма относительно любого морфизма есть регулярный эпиморфизм [82, p. 41].

При этом *регулярный эпиморфизм* – это морфизм, который представим как коуравнитель.

4.10.3. Точное пополнение категорий

Определение 4.22. Регулярная категория $\mathcal{K}$ называется *точной*, если для каждого отношения эквивалентности $r : R \longrightarrow X \times X$ пара морфизмов $pr_1 \circ r, pr_2 \circ r : R \longrightarrow X$ является ядерной парой для r (и значит, имеет коуравнитель) [82, p. 46].

В случае, когда рассматриваемая категория не является точной, к ней применяется операция точного пополнения.

Определение 4.23. *Точным пополнением* регулярной категории $\mathcal{K}$ называется категория $EC(\mathcal{K})$, задаваемая следующим образом:

1) объекты есть пары (X, ρ), где X – объект категории $\mathcal{K}$ и ρ – отношение эквивалентности на X;

2) морфизмы $(X, \rho) \longrightarrow (Y, \sigma)$ – это отношения $\tau : T \rightarrowtail X \times Y$ такие, что

$$\sigma.\rho.\tau \subseteq \tau,$$
$$\rho \subseteq \tau^{\circ}.\tau,$$
$$\tau.\tau^{\circ} \subseteq \sigma;$$

3) тождественный морфизм $(X, \rho) \longrightarrow (X, \rho)$ есть отношение $\rho : R \rightarrowtail X \times X$;

4) композиция морфизмов задается композиционным отношением [82, p. 46].

Как видим, пополнение категории $\mathcal{K}$ происходит посредством факторизации ее объектов, основанной на введении в объект некоторого отношения эквивалентности. В результате к объектам категории $\mathcal{K}$ добавляются новые дополнительные объекты (фактор-объекты), полученные факторизацией по отношению эквивалентности объектов категории $\mathcal{K}$. Так происходит пополнение категории $\mathcal{K}$ новыми объектами. Точное пополнение не всегда дает топос, и неизвестно, в каком случае эта процедура ведет к топосу. В диссертации M. Menni [86] дается характеристика конечно полных категорий, чьи точные пополнения являются топосами.

Теорема 4.4. *Если $\mathcal{K}$ регулярная категория, то категория $EC(\mathcal{K})$ является точной.* ■

«Грубо говоря, точная категория – это категория с хорошо ведущими себя факторизациями (фактор-объектами) по внутренним отношениям эквивалентности. Каждая точная категория является регулярной, т. е. с хорошо ведущими себя образами факторизаций (для данного $f : A \longrightarrow B$ можно ввести отношение эквивалентности на A, принимая, что $a \sim a'$, если $f(a) = f(a')$; результат факторизации $A/_{\sim}$ по отношению эквивалентности $\sim$ есть образ факторизации для f).

Смысл точного пополнения в том, что оно дает универсальный способ получения из регулярной категории $\mathcal{K}$ точной категории $EC(\mathcal{K})$. В самом деле, если мы хотим знать, какими должны быть объекты точного пополнения, то мы это знаем – они должны быть результатом факторизации по отношениям эквивалентности в $\mathcal{K}$; поэтому мы должны использовать имеющиеся отношения эквивалентности в $\mathcal{K}$, чтобы получить "точное пополнение". Морфизмы между фактор-объектами двух отношений эквивалентности могут быть охарактеризованы в терминах отношений эквивалентности – они являются "цельными" и "функциональными" бинарными отношениями относительно данных отношений эквивалентности; поэтому можем использовать это, чтобы определить морфизмы в точном пополнении» (Mike Shulman, 2012).

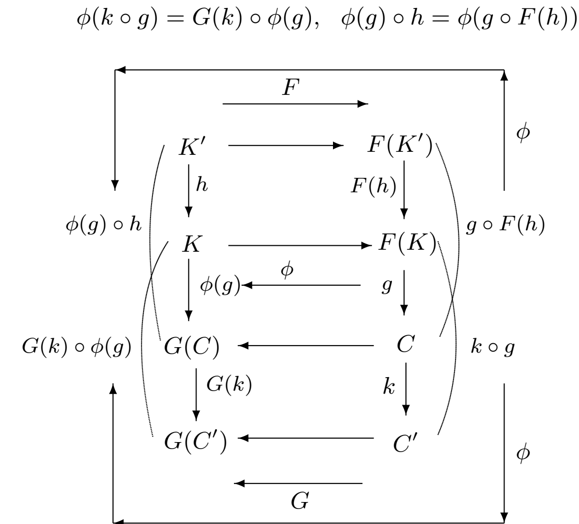
4.11. Геометрический морфизм между топосами

Рассмотрим две категории $\mathcal{K}$ и $\mathcal{C}$ и «встречные» функторы между ними $F : \mathcal{K} \longrightarrow \mathcal{C}$ и $G : \mathcal{C} \longrightarrow \mathcal{K}$. Помимо их взаимообратности, влекущей эквивалентность категорий, возможно более тонкое отношение между ними, называемое *сопряженностью* и широко используемое в теории категорий.

4.11.1. Сопряжение функторов

Сопряжение двух функторов $F : \mathcal{K} \longrightarrow \mathcal{C}$ и $G : \mathcal{C} \longrightarrow \mathcal{K}$ означает наличие отображения ϕ, которое каждому морфизму $g : F(K) \longrightarrow C$, $K \in \mathcal{K}, C \in \mathcal{C}$ ставит в соответствие морфизм $\phi(g) : K \longrightarrow G(C)$ так, что справедивы равенства

$$\phi(k \circ g) = G(k) \circ \phi(g), \quad \phi(g) \circ h = \phi(g \circ F(h))$$



для любых морфизмов $k : C \longrightarrow C'$ и $h : K' \longrightarrow K$, где $K' \in \mathcal{K}$, $C' \in \mathcal{C}$.

Говорят, F – *левосопряженный* к G, а G – *правосопряженный* к

F. Символически: $F \dashv G$ и

$$\frac{\mathcal{K} \xrightarrow{\phi(g)} G\mathcal{C}}{F\mathcal{K} \xrightarrow{g} \mathcal{C}}.$$

4.11.2. Определения геометрического морфизма

Определение 4.24. *Геометрический морфизм* $F : \mathcal{K} \longrightarrow \mathcal{E}$ между топосами есть пара сопряженных функторов $F_* : \mathcal{K} \longrightarrow \mathcal{E}$ и $F^* : \mathcal{E} \longrightarrow \mathcal{K}$, т. е. $F_* \dashv F^*$, таких, что F^* сохраняет все конечные пределы.

В этом случае F_* называется прямым образом F, а F^* – обратным образом геометрического морфизма.

Геометрический морфизм $F : \mathcal{K} \longrightarrow \mathcal{E}$ называют *сюръекцией*, если обратный образ F^* точен (faithful), т. е. для любых $\mathcal{E}$-объектов A, B операция

$$Hom_{\mathcal{E}}(A, B) \rightarrow Hom_{\mathcal{K}}(F^*(A), F^*(B)), \quad f \rightarrow F^*(f),$$

является инъекцией.

4.12. Кванторы

Определим функторы, которые являются аналогами кванторов в теории множеств.

Пусть дана категория $\mathcal{K}$ с обратными образами и $\mathcal{K}$-объект X. Определим категорию $\mathcal{K}/X$, объектами которой являются $\mathcal{K}$-морфизмы $f : Y \longrightarrow X$, а морфизмами из $f : Y \longrightarrow X$ в $g : Z \longrightarrow X$ $\mathcal{K}$-морфизм $h : Y \longrightarrow Z$ такой, что $f = g \circ h$. Если $\mathcal{K}$ – топос, то $\mathcal{K}/X$ – топос.

Рассмотрим морфизм $f : X \longrightarrow Y$. Тогда существует функтор

$$f^* : \mathcal{K}/Y \rightarrow \mathcal{K}/X,$$

называемый *функтором обратного образа.*

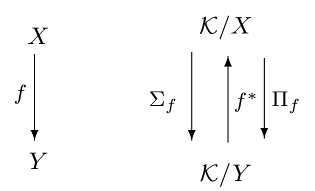
В самом деле, если дан морфизм $g : A \longrightarrow Y$, то существует обратный образ

$$\begin{array}{ccc} X \times_Y A & \longrightarrow & A \\ {\scriptstyle f^*(g)}\downarrow & & \downarrow{\scriptstyle g} \\ X & \underset{f}{\longrightarrow} & Y \end{array}$$

который дает морфизм $f^*(g) : X \times_Y A \longrightarrow X$.

Теорема 4.5. *Пусть $\mathcal{K}$ категория с обратными образами и $f : X \longrightarrow Y$ ее морфизм. Тогда функтор $f^* : \mathcal{K}/Y \longrightarrow \mathcal{K}/X$ имеет левый и правый сопряженные функторы Σ_f и Π_f*

$$\Sigma_f \dashv f^* \dashv \Pi_f.$$



Категория $Sub(X)$ является полной подкатегорией категории $\mathcal{K}/X$ (см.: [15, с. 164]), а функторы f^* и Π_f сохраняют конечные объекты и мономорфизмы. Поэтому f^* и Π_f можно ограничить на $Sub(Y)$, $Sub(X)$ и получить функторы

$$f^{-1} : Sub(Y) \longrightarrow Sub(X) \quad \text{и} \quad \forall_f : Sub(X) \longrightarrow Sub(Y),$$

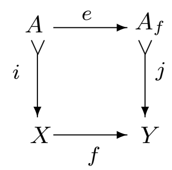
$$f^{-1} \dashv \forall_f,$$

первый из которых был определен в § 4.6.3.

Функтор $\forall_f$ – аналог квантора $\forall$ в теории множеств **Sets**.

Можно найти левосопряженный к f^{-1} функтор $\exists_f$, если взять мономорфизм $i : A \rightarrowtail X$, т. е. объект категории $Sub(X)$ и найти эпи-моно-разложение $A \rightarrowtail^j S_f \twoheadrightarrow^e Y$ композиции $A \rightarrowtail^i X \xrightarrow{f} Y$ (см. теорему 4.1), при котором находится объект A_f такой, что коммутативна диаграмма

■



Таким образом, найден функтор

$$\exists_f : Sub(X) \longrightarrow Sub(Y),$$

$$\exists_f A =^{def} A_f,$$

$$\exists_f \dashv f^{-1}.$$

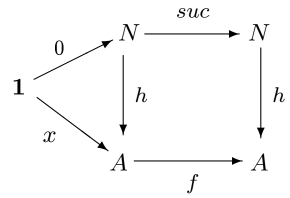
Функтор $\exists_f$ – аналог квантора $\exists$ в теории множеств **Sets**.

4.13. Числовые объекты

4.13.1. Объект натуральных чисел

В топосах могут существовать объекты, которые выполняют роль множества натуральных чисел $\mathbb{N}$ в **Sets**.

Объект натуральных чисел в топосе $\mathcal{E}$ – это объект N и морфизмы $\mathbf{1} \xrightarrow{0} N \xrightarrow{suc} N$ такие[6], что для любого объекта A и морфизмов $\mathbf{1} \xrightarrow{x} A \xrightarrow{f} A$ имеется ровно один морфизм $h : N \longrightarrow A$, для которых диаграмма



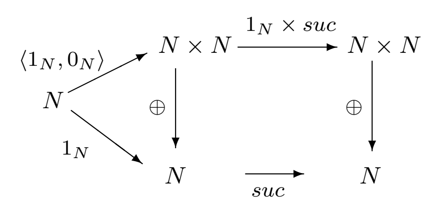
коммутативна.

В **Sets** морфизм suc – это функция $suc(n) = n + 1$.

Теорема 4.6. *Каждый топос Гротендика имеет объект натуральных чисел.* ∎

Для объекта N определяется операция сложения как морфизм $\oplus : N \times N \longrightarrow N$, для которого коммутативна диаграмма:

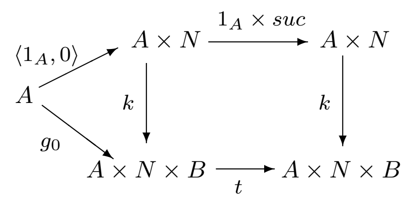
[6] Suc – сокращение англ. successor (последующий).



где 0_N есть композиция $! : N \longrightarrow \mathbf{1}$ и $0 : \mathbf{1} \longrightarrow N$.

Умножение $\otimes : N \times N \longrightarrow N$ определено в [9, с. 357].

Теорема 4.7. (Примитивная рекурсия в топосе). *Для каждых* $g_0 : A \longrightarrow A \times N \times B$ *и* $t : A \longrightarrow A \times N \times B \longrightarrow B$ *существует единственный морфизм* $k : A \times N \longrightarrow A \times N \times B$ *такой, что коммутативна диаграмма*



■

В топосе **Sets** этой диаграмме соответствует схема примитивной рекурсии

$$\begin{cases} f(a,0) = g(a), \\ f(a,n+1) = h(a,n,f(a,n)), \end{cases}$$

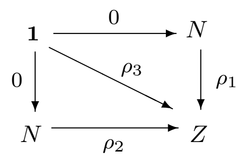
где $f : A \times N \longrightarrow B$, $g : A \longrightarrow B$, $h : A \times N \times B \longrightarrow B$. Действительно, достаточно рассмотреть отображения $k(a,n) = (a,n,f(a,n))$, $g_0(a) = (a,0,g(a))$, $t(a,n,b) = (a,n+1,h(a,n,b))$, $suc(n) = n+1$.

Имея схему примитивной рекурсии в топосе, можно развивать теорию вычислимых морфизмов.

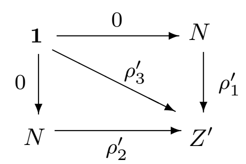
4.13.2. Объект целых чисел

Рассмотрим топос $\mathcal{K}$ с объектом натуральных чисел. В таких топосах можно построить объект целых чисел.

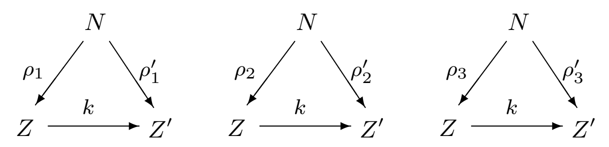
Определение 4.25. *Объект целых чисел* Z – это объект, для которого коммутативна диаграмма



где $\rho_1, \rho_2 : N \rightarrowtail Z$, $\rho_3 : \mathbf{1} \rightarrowtail Z$ – морфизмы, и если существует Z с коммутативной диаграммой



где $\rho_1', \rho_2' : N \rightarrowtail Z'$, $\rho_3' : \mathbf{1} \rightarrowtail Z$ – морфизмы, то существует единственный морфизм $k : Z \longrightarrow Z'$ такой, что три диаграммы

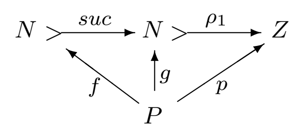


коммутативны.

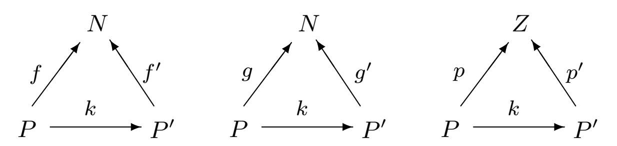
Объект положительных целых чисел в топосе $\mathcal{K}$ – это подобъект $p : P \rightarrowtail Z$, который является пределом диаграммы

$$N \overset{suc}{\rightarrowtail} N \overset{\rho_1}{\rightarrowtail} Z,$$

т. е. имеет место коммутативная диаграмма

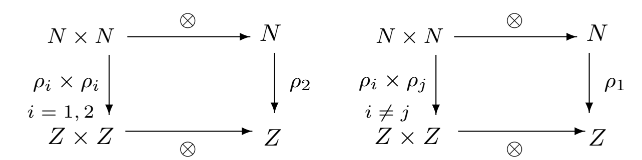


и если существует подобъект $p' : P' \rightarrowtail Z$ и морфизмы $f' : P \longrightarrow N$, $g' : P \rightarrowtail N$, удовлетворяющие этой же диаграмме, то существует единственный морфизм $k : P' \longrightarrow P$ такой, что три диаграммы



коммутативны.

Операция умножения в объекте целых чисел – это единственный морфизм $\otimes : Z \times Z \longrightarrow Z$, для которого коммутативны диаграммы



Рассмотрим проекции

$$pr_1 : (Z \times P) \times (Z \times P) \longrightarrow Z \times P,$$

$$pr_2 : (Z \times P) \times (Z \times P) \longrightarrow Z \times P,$$

$$p_1 : (Z \times P) \longrightarrow Z,$$

$$p_2 : (Z \times P) \longrightarrow P$$

и введем морфизмы π_{14}, π_{32}, которые используем в следующем разделе:

$$\begin{array}{ccc} Z \times P & \xrightarrow{\;p_1\;} & Z \\ \uparrow{\scriptstyle pr_1} & & \uparrow{\scriptstyle p_1} \\ (Z \times P) \times (Z \times P) & \xrightarrow{\;\pi_{14}\;} & Z \times P \\ \downarrow{\scriptstyle pr_2} & & \downarrow{\scriptstyle p_2} \\ Z \times P & \xrightarrow[\;p_2\;]{} & P \end{array}$$

$$
\begin{array}{ccc}
Z \times P & \xrightarrow{\quad p_1 \quad} & Z \\
{\scriptstyle pr_2}\uparrow & & \uparrow{\scriptstyle p_1} \\
(Z \times P) \times (Z \times P) & \xrightarrow{\ \pi_{32}\ } & Z \times P \\
{\scriptstyle pr_1}\downarrow & & \downarrow{\scriptstyle p_2} \\
Z \times P & \xrightarrow[\quad p_2 \quad]{} & P
\end{array}
$$

$$
\pi_{14} = \langle p_1 \circ pr_1, p_2 \circ pr_2 \rangle, \qquad \pi_{32} = \langle p_1 \circ pr_2, p_2 \circ pr_1 \rangle. \tag{4.2}
$$

4.13.3. Объект рациональных чисел

Рассмотрим топос $\mathcal{K}$ с объектом натуральных чисел. В таких топосах можно построить объект рациональных чисел.

Определение 4.26. *Объект рациональных чисел* – это объект Q с коуравнителем

$$
E \overset{pr_1 \circ i}{\underset{pr_2 \circ i}{\rightrightarrows}} Z \times P \xrightarrow{\ q\ } Q,
$$

где E – объект, удовлетворяющий уравнителю

$$
E \xrightarrow{\ i\ } (Z \times P) \times (Z \times P) \overset{\pi_{14}}{\underset{\pi_{32}}{\rightrightarrows}} Z \times P \xrightarrow{1_Z \times p} Z \times Z \xrightarrow{\ \otimes\ } Z,
$$

где морфизмы π_{14}, π_{32} даны в (4.2).

4.14. Внутренняя логика топоса. Язык Митчела–Бенабу

При конструировании различных объектов в топосе используются различные категорные операции, например конечные пределы, обратные образы, экспоненциалы. Примером категорного определения являются определения объектов натуральных, целых и рациональных чисел, данные в § 4.13. Но можно ввести специальный

логический язык, который подобен способу образования множеств в категории **Sets**, а именно: объекты определяются как

$$\{x \mid \phi(x)\},$$

где ϕ – формула вводимого в топос логического языка.

Иначе говоря, объекты в топосе определяются как своего рода «множества», состоящие из «элементов» x, обладающих некоторым свойством ϕ. Такой подход интуитивно естественен, поскольку теория топосов – это «обобщенные вселенные» множеств. Что важно: с помощью такого языка можно писать формулы на ставшем привычным для математиков за сто лет развития математики теоретико-множественном языке, но которые, естественно, уже имеют иную интерпретацию в форме категорных объектов, морфизмов и диаграмм.

4.14.1. Описания языка Митчела–Бенабу

Логический язык, о котором мы говорим, называется теоретико-множественным *языком Митчела–Бенабу*. Он соответствует тому, что именуют *внутренней логикой топоса.*

Поскольку в топосе есть различного типа объекты, скажем, объект натуральных чисел N, объект рациональных чисел Z, то переменные $x, x', x'', ...$ и константы $c, c', c'', ...$ – термы, соответственно «пробегающие» и «принадлежащие» разнотипным объектам, – должны помечаться как термы разных типов.

Определение 4.27. *Язык $L_{\mathcal{K}}$ внутренней логики топоса $\mathcal{K}$* состоит из следующих символов:

1. *Типы* $X, Y, ...$ Соответствуют (представляют) объектам топоса $\mathcal{K}$.

2. *Термы* каждого типа конструируются следующим образом:

- для каждого типа X имеется запас переменных $x, x', x'', ...$; символически это записывается как формула $x \in X$;
- для каждого морфизма $f : X \longrightarrow Y$ топоса $\mathcal{K}$ и каждого терма τ типа X имеется терм $f(\tau)$ типа Y; можно писать $x \in X$, $f(\tau) \in Y$, $\langle x, f(x)\rangle \in X \times Y$;
- если заданы термы τ типа X и σ типа Y, то имеется терм $\langle\tau, \sigma\rangle$ типа $X \times Y$. Можно писать $\langle\tau, \sigma\rangle \in X \times Y$.

3. *Формулы* – это термы типа Ω. Более конкретно, имеется:

- бинарный предикат $=_X$ сигнатуры (X, X) для каждого типа X;
- бинарный предикат $\in_X$ сигнатуры (X, Ω^X) для каждого типа X (определяется аналогично с использованием отображения оценки $X \times \Omega^X \to \Omega$);
- логические связки $\vee, \wedge, \Rightarrow, \neg$ (индуцируемые операциями гейтинговой алгебры на Ω, см. § 4.6.4).

4. *Кванторы и описания.* Пусть ϕ – формула и x – переменная. Тогда:

- $\{x|\phi\}$ – терм типа Ω^X;
- $\forall x.\phi$ и $\exists x.\phi$ – формулы;
- $\lambda x.\phi$ – терм типа Z^X, где x – переменная типа X, ϕ – типа Z, отвечающий морфизму $X \times U \to Z$.

Понятия свободного и связанного вхождения переменных в терм $L_{\mathcal{K}}$ определяются как обычно, т. е. переменная свободна, если она не связана квантором или оператором описания.

4.14.2. Интерпретация внутреннего логического языка Митчела–Бенабу

Интерпретация в топосе $\mathcal{K}$ формул, записанных на языке Митчела–Бенабу, дается в форме диаграмм, характеризующих переменные как объекты этих диаграмм, связанных морфизмами (см.: [15, с. 173–174] или [83, p. 298–301]).

1. Тип X терма интерпретируем в топосе как объект X.

2. Переменную x типа X, или теоретико-множественное выражение $x \in X$, интерпретируем в топосе как морфизм $1_X : X \to X$; терм $f(x)$, или выражение $f(x) \in Y$, интерпретируем как морфизм $f : X \to Y$.

3. Терм $t(x_1, ..., x_n)$ типа Y, или выражение $t(x, ..., x_n) \in Y$, со свободными переменными $x_1, ..., x_n \in X_1 \times ... \times X_n$ интерпретируется как морфизм

$$[t] : X_1 \times ... \times X_n \to Y.$$

4. Формула $\tau =_X \sigma$, где τ, σ – термы типа X. При интерпретации имеем два равных морфизма $[\![\tau]\!], [\![\sigma]\!]$, т. е. $[\![\tau]\!] = [\![\sigma]\!]$. Если x, x' –

переменные типа X, то для них $x =_X x'$ интерпретируется как один и тот же морфизм $1_X : X \to X$.

5. Если термы $\tau \in Y$ и $\sigma \in Z$ имеют одну и ту же свободную переменную $x \in X$, то $\langle \tau, \sigma \rangle$ интерпретируется как морфизм

$$\langle [\![\tau]\!], [\![\sigma]\!] \rangle : X \longrightarrow Y \times Z.$$

Если σ, τ – термы типа X, то $(\sigma =_X \tau)$ есть $\delta_X(\langle \sigma, \tau \rangle)$, где $\delta_X : X \times X \longrightarrow \Omega$ классифицирует диагональный подобъект $\Delta_X : X \rightarrowtail X \times X$,

$$\begin{array}{ccc} X & \overset{\Delta_X}{\rightarrowtail} & X \times X \\ {\scriptstyle !}\downarrow & & \downarrow{\scriptstyle \delta_X} \\ \mathbf{1} & \xrightarrow{\top} & \Omega \end{array}$$

6. Формула, т. е. терм $\phi(x)$ типа Ω, или теоретико-множественное выражение $\phi(x)$, со свободной переменной $x \in X$ интерпретируется как морфизм

$$[\![\phi]\!] : X \longrightarrow \Omega.$$

Этот морфизм классифицирует подобъект $||\phi|| \rightarrowtail X$

$$\begin{array}{ccc} ||\phi|| & \rightarrowtail & X \\ {\scriptstyle !}\downarrow & & \downarrow{\scriptstyle [\![\phi]\!]} \\ \mathbf{1} & \xrightarrow{\top} & \Omega \end{array}$$

7. Даны формулы $\phi(x)$ и $\psi(x)$ со свободной переменной $x \in X$. Тогда формула $\phi \vee \psi$ интерпретируется как композиция морфизмов

$$[\![\phi \vee \psi]\!] : X \xrightarrow{\langle [\![\phi]\!], [\![\psi]\!] \rangle} \Omega \times \Omega \xrightarrow{\cup} \Omega,$$

$$[\![\phi \wedge \psi]\!] : X \xrightarrow{\langle [\![\phi]\!], [\![\psi]\!] \rangle} \Omega \times \Omega \xrightarrow{\cap} \Omega,$$

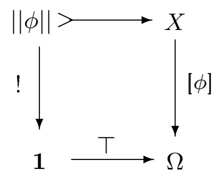
$$[\![\neg\phi]\!] : X \xrightarrow{[\![\phi]\!]} \Omega \xrightarrow{\neg} \Omega,$$

где морфизмы $\Omega \times \Omega \xrightarrow{\cup} \Omega$, $\Omega \times \Omega \xrightarrow{\cap} \Omega$ и $\neg : \Omega \longrightarrow \Omega$ описаны в § 4.6.4.

8. **Интерпретация $\{\mathbf{x} \mid \phi(\mathbf{x})\}$.** Если $\phi(x)$ формула со свободной переменной $x \in X$, то

$$\{x \mid \phi(x)\}$$

интерпретируется как подобъект $||\phi|| \rightarrowtail X$, который классифицируется морфизмом $[\![\phi]\!]$



9. **Интерпретация формул с кванторами**. Пусть $\phi(x, y)$ предикат на $X \times Y$ с интерпретацией

$$[\![\phi]\!] : X \times Y \longrightarrow \Omega,$$

которая классифицирует подобъект $||\phi|| \rightarrowtail X \times Y$.

Рассмотрим проекцию $\pi : X \times Y \longrightarrow Y$, функтор обратного образа

$$\pi^{-1} : Sub(Y) \longrightarrow Sub(X \times Y)$$

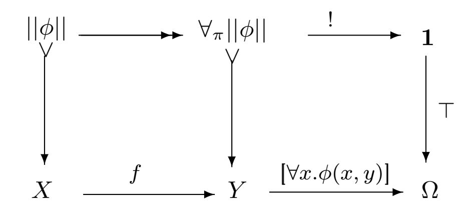
и его лево- и правосопряженные функторы (см. § 4.12)

$$\forall_\pi : Sub(X \times Y) \longrightarrow Sub(Y) \text{ и } \exists_\pi : Sub(X \times Y) \longrightarrow Sub(Y).$$

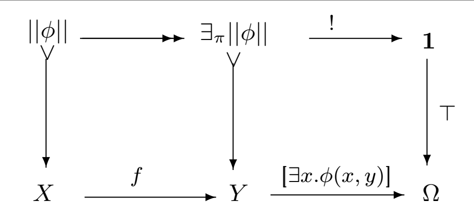
Тогда формулы $\forall x \in X.\phi(x, y)$ и $\exists x \in X.\phi(x, y)$ интерпретируются как морфизмы

$$[\![\forall x.\phi(x, y)]\!], \quad [\![\exists x.\phi(x, y)]\!] : Y \longrightarrow \Omega,$$

которые вместе с мономорфизмами $\forall_\pi ||\phi|| \rightarrowtail Y$, $\exists_\pi ||\phi|| \rightarrowtail Y$ удовлетворяют соответственно диаграмме (см. конец § 4.12)



и диаграмме



в которых правые квадраты являются декартовыми.

Определение 4.28. Дана формула ϕ на X, т. е. формула с одной свободной переменной x типа X. Она *истинна в топосе* $\mathcal{E}$,

$$\mathcal{E} \models \phi,$$

если

$$(\llbracket\phi\rrbracket : X \to \Omega) = (X \xrightarrow{!} \mathbf{1} \xrightarrow{\top} \Omega),$$

т. е.

$$||\phi|| = X \quad \text{и} \quad (||\phi|| \rightarrowtail X) = 1_X,$$

или имеем декартов квадрат

$$\begin{array}{ccc} ||\phi|| & \stackrel{1_X}{\rightarrowtail} & X \\ {\scriptstyle !}\downarrow & & \downarrow{\scriptstyle [\phi] = \top} \\ \mathbf{1} & \xrightarrow{\top} & \Omega \end{array}$$

4.15. Объект действительных чисел

В теории множеств **Sets** множество действительных чисел $\mathbb{R}$ определяется с помощью сечений Дедекинда (L, U) формулы

$$\mathbb{R} = \{\langle L, U\rangle \in 2^Q \times 2^Q \mid \phi(L, U)\},$$

где (L, U) – пара непересекающихся подмножеств (L, U) множества рациональных чисел Q; $\phi(L, U)$ – конъюнкция следующих условий:

$$
\begin{aligned}
&1)\ \exists q \in Q.q \in L,\\
&\quad \exists r \in Q.r \in U;\\
&2)\ \forall q, r \in Q.q < r \wedge r \in L \Rightarrow q \in L,\\
&\quad \forall q, r \in Q.r < q \wedge r \in U \Rightarrow q \in U;\\
&3)\ \forall q \in Q.q \in L \Rightarrow \exists r \in Q(r \in L \wedge q < r);\\
&4)\ \forall q \in Q.q \in U \Rightarrow \exists r \in Q(r \in U \wedge r < q);\\
&5)\ \forall q, r \in Q.q < r \Rightarrow (q \in L \vee r \in U);\\
&6)\ L \cap U = \emptyset.
\end{aligned}
\tag{4.3}
$$

Язык Митчела–Бенабу также позволяет написать формулу $\phi(L, U)$, понимая под Q тип Q, сопоставленный объекту рациональных чисел. Следовательно, на языке Митчела–Бенабу объект действительных чисел можно определить посредством следующей формулы:

$$R = \{\langle L, U\rangle \in \Omega^Q \times \Omega^Q \mid \phi(L, U)\}.$$

Интерпретация этой формулы приведет к набору диаграмм, которые и определяют объект действительных чисел, подобно тому, как были определены числовые объекты N, Z и Q.

4.16. У. Ловер


Уильям Ловер

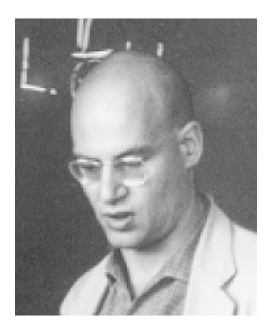
Уильям Ловер (Francis William Lawvere, род. 9 февраля 1937) – американский математик. Известен своими работами по теории категорий, теории топосов и философии математики. Получил докторскую степень (Ph.D.) в 1963 году под руководством Эйленберга. В период написания диссертации Ловер провел год в Университете Беркли, посещая лекции Тарского и Скотта по теории моделей и теории множеств.

В 1968–1969 годах Ловер предложил элементарные (касающиеся логики первого порядка) аксиомы топосов, обобщив понятие топоса Гротендика. Он ввел понятие *элементарного топоса* как ответ на вопрос, каким условиям должна удовлетворять категория, чтобы быть «в

сущности такой же», как **Sets**. Теория категорий через понятие элементарного топоса достигла успеха в аксиоматизации теории множеств, и в результате возникло совершенно новое категорное основание математики [9, с. 15].

Среди аспирантов Ловера, получивших ученую степень доктора, Marta Bunge, Anders Kock, Philip Mulry.

4.17. А. Гротендик



А. Гротендик

Александр Гротендик (Alexander Grothendieck, 28 марта 1928, Берлин – 13 ноября 2014, Сен-Жирон) – французский математик. Лицо без гражданства. Говорил по-русски. Входил в группу математиков, которые выступали под псевдонимом «Николя Бурбаки». Работал в области функционального анализа, теории категорий и, в основном, в алгебре и алгебраической геометрии. Лауреат премии Филдса (1966).

Отец: Александр Шапиро – эмигрант из России, участник революции 1905 года, был приговорен к смертной казни, замененной ввиду несовершеннолетия лишением свободы. Мать: Иоганна (Ханка) Гротендик – немка. Родители были анархистами. Принимали активное участие в Гражданской войне в Испании на стороне республиканцев. Отец погиб (1942) в Освенциме.

Гротендик отказался приехать в 1966 году на XV Московский математический конгресс, где ему должны были вручить медаль Филдса, поскольку считал, что СССР не является свободным государством. Но поехал в коммунистический Вьетнам, чтобы читать лекции в эвакуированном в джунгли Ханойском университете, несмотря на американские бомбардировки. В США Гротендик не был, заявив, что «американский народ, гордясь своей свободой, смеет подавлять свободу других народов».

Закончил университет в Монпелье (Франция). Продолжил образование в Париже. Учителя: Жан Дьёдонне и Лоран Шварц.

В 1960 году Гротендик открыл класс категорий, который впоследствии назвал топосами.

Глава 5

Эффективный топос

Один хороший пример стоит массы общих мест.

Хайлэнд [105].

Эффективный топос $\mathcal{E}\!f\!f$ придуман с целью построения интерпретации конструктивной арифметики, основанной на понятии рекурсивной реализации Клини, осуществляющей эффективное, вычислимое установление истинности формул и утверждений.

Эффективный топос – прекрасный пример синтетического мира, в котором всё вычислимо [106].

Эффективный топос введен в теорию рекурсивной реализуемости Хайлэндом (1980, [73]). Понятие «вычисление» встроено в организацию эффективного топоса; его объекты и морфизмы снабжены вычислимыми структурами. Все функции $f : \mathbb{N} \longrightarrow \mathbb{N}$ при интерпретации их в эффективном топосе оказываются вычислимыми, а тезис Чёрча справедлив. В действительности, и все функции между объектами, построенными из натуральных чисел, включающими и рациональный, и вещественный, рекурсивны [85, p. 229]. В частности, рекурсивны при интерпретации все функции $g : \mathbb{R} \longrightarrow \mathbb{R}$. Поэтому такие объекты пригодны для моделирования типов данных в языках программирования, а эффективный топос $\mathcal{E}\!f\!f$ – естественная основа для рекурсивной теории высокого порядка [85, p. 229].

Эффективный топос не является топосом Гротендика. Внутренняя логика 1-го порядка эффективного топоса совпадает с поняти-

ем реализуемости Клини [67].

«Эффективный топос является странной вещью, понимание его логики редко́ даже среди специалистов по топосам; и хотя к настоящему времени издано много книг по теории топосов[1], <в том числе и учебников для студентов факультетов компьютерных наук>, ни в одной из них эффективный топос не рассматривается, как тот элемент, который заслуживает особого внимания» (J. van Oosten, [105]). Отчасти такую ситуацию объяснил сам первооткрыватель эффективного топоса Хайлэнд, сказав, что еще «слишком рано для книг по реализуемости», хотя трудно понять, что он имел в виду – сложность теории топосов для теоретиков в области вычислимости или далекость специалистов по топосам от проблем вычислимости.

5.1. Объекты топоса $\mathcal{E}\!f\!f$

Объект топоса $\mathcal{E}\!f\!f$ – это пара (X, R), где X – множество и $R \in P(\mathbb{N})^{X \times X}$ – $P(\mathbb{N})$-значный предикат, т. е. каждое (значение) $R(x, y)$ есть подмножество[2] множества натуральных чисел $\mathbb{N}$. Любое натуральное число $n \in R(x, y)$ – это код информации, устанавливающий «$x = y$».

Предполагается, что выполняются следующие две аксиомы: существуют частично рекурсивные функции σ, τ такие, что

1) (симметричность) для $x, y \in X$ и $n \in R(x, y)$, $\sigma(n)\downarrow$ и $\sigma(n) \in R(y, x)$;

2) (транзитивность) для $x, y, z \in X$, $n \in R(x, y)$ и $m \in R(y, z)$, $\tau(n, m)\downarrow$ и $\tau(n, m) \in R(x, z)$.

Здесь используется обозначение $\phi(n)\downarrow$ для «n принадлежит области определения частично рекурсивной функции ϕ».

Мы не требуем аксиомы рефлексивности. Множество $R(x, x)$ может быть даже пустым для некоторого $x \in X$.

Будем называть $R(x, x)$ предикатом существования для (X, R), поскольку элементы $R(x, x)$ – это «свидетели» существования x. Будем использовать обозначение $E(x)$ для $R(x, x)$.

Конечный объект 1. Пусть $\{*\}$ одноэлементное множесто. Рассмотрим следующее отношение R на $\{*\}$: $R(\{*\}, \{*\}) = \mathbb{N}$. Тогда $(\{*\}, R)$ – терминальный объект $\mathbf{1}$ в $\mathcal{E}\!f\!f$.

---

[1] На русском языке таких книг только две – [9], [15].

[2] В литературе для множества $R(x, y)$ используется и другое обозначение – $[x =_X y]$ (см., например: [105]). Вместо (X, R) пишут $(X, =_X)$.

5.2. Морфизмы топоса $\mathcal{E}ff$

Пусть (X, R) и (Y, S) два объекта в $\mathcal{E}ff$. Назовем *функциональным отношением* из (X, R) в (Y, S) такую функцию $F : X \times Y \to P(\mathbb{N})$, для которой существуют частично рекурсивные функции ex, st, sv, tl, удовлетворяющие следующим условиям:

1) для $n \in R(x', x)$, $m \in S(y, y')$ и $k \in F(x, y)$, $ex(n, m, k)\downarrow$ и $ex(n, m, k) \in F(x', y')$ (ex – экстенсионал-функция);

2) для $n \in F(x, y), st_1(n) \in E(x)$ и $st_2(n) \in E(y)$, $st(n) = \langle st_1(n), st_2(n)\rangle$ (st – strict-функция);

3) для $n \in F(x, y)$ и $m \in F(x, y')$, $sv(n, m)\downarrow$ и $sv(n, m) \in S(y, y')$ (sv – single-valued-функция);

4) для $n \in E(x)$, $tl(n)\downarrow$ и $tl(n) \in \bigcup_{y\in Y} F(x, y)$ (tl – total-функция).

Функциональные отношения из (X, R) в (Y, S) образуют предпорядок: $F \leq G$, если для некоторой частично рекурсивной функции e, как только $n \in F(x, y)$, то $e(n) \in G(x, y)$ (для всех x, y).

Морфизм $(X, R) \longrightarrow (Y, S)$ в $\mathcal{E}ff$ – это класс эквивалентности функциональных отношений из $F : (X, R)$ в (Y, S) относительно отношения эквивалентности $(F \leq G) \wedge (G \leq F)$.

Функциональное отношение, представляющее морфизм $f : (X, R) \longrightarrow (Y, S)$, будем обозначать как $F_f : (X, R) \longrightarrow (Y, S)$.

Таким образом, морфизм – это бинарный предикат $F(\bar{x}, \bar{y})$, истинность которого может быть установлена, *вычислена* с помощью инструкций, гёделевские коды которых – это подмножество $F \in P(\mathbb{N})$.

Для каждого объекта (X, R) R является функциональным отношением (X, R) в себя.

Композиция морфизмов определяется следующим образом. Если F представляет морфизм $f : (X, R) \longrightarrow (Y, S)$ и G представляет морфизм $g : (Y, S) \longrightarrow (Z, T)$, то композиция $g \circ f$ есть морфизм

$$(G \circ F)(x, z) = \bigcup_{y\in Y} \{\langle a, b\rangle |\ a \in F(x, y) \ \wedge\ b \in G(y, z)\}.$$

5.3. Конечные произведения

Конечное произведение в $\mathcal{E}ff$ задается следующим образом. Произведение $(X, R)\times(Y, S)$ объектов (X, R) и (Y, S) есть $(X\times Y, R\times S)$,

где $R \times S$ определяется как

$$(R \times S)((x,y),(x',y')) = R(x,x') \times S(y,y') =$$

$$= \{n = \langle n_1, n_2\rangle |\ n_1 \in R(x,x') \ \wedge \ n_2 \in S(y,y')\}.$$

Здесь используется примитивно рекурсивное кодирование для пар и последовательностей $\langle \cdot, \cdot \rangle$; когда рассматривается число n – номер (код) пары, то в этом случае пишем n_1, n_2 для ее компонент: $n = \langle n_1, n_2\rangle$.

5.4. Экспоненциал

Для объектов (X,R) и (Y,S) *экспоненциал* $(Y,S)^{(X,R)}$ – это объект (Φ, T), где Φ есть множество функций $X \times Y \longrightarrow P(\mathbb{N})$.

Для того чтобы определить T, мы, во-первых, определяем для $F \in \Phi$ множество $E(F)$ элементов, которые проверяют, что F «существует»; другими словами: есть функция $(X,R) \longrightarrow (Y,S)$. Мы определяем $E(F)$ как множество всех четверок $n = (n_1, n_2, n_3, n_4)$ таких, что n_1 есть индекс частично рекурсивной функции ex, которая проверяет, что F есть экстенсионал (см. определение морфизмов в $\mathcal{E}\!f\!f$), и подобно n_2, n_3, n_4 – индексы соответственно для strict-, single-valued- и total-функций.

Пусть $F, G \in P(\mathbb{N})^{X \times Y}$. Обозначаем через $[F \leq G]$ множество индексов n таких, что для всех x, y и всех $m \in F(x,y)$, $n(m){\downarrow}$ и $n(m) \in G(x,y)$. Полагаем

$$T(F,G) = \{\langle k,l,m\rangle |\ k \in E(F), l \in E(G), m \in [F \leq G]\}.$$

5.5. Классификатор

$$\Omega = (P(\mathbb{N}), \cdot \Leftrightarrow \cdot)$$

Здесь $n \in (A \Leftrightarrow B)$ есть пара $\langle n_1, n_2\rangle$ индексов частично рекурсивных функций таких, что для каждого $a \in A$, $n_1(a){\downarrow}$ и $n_1(a) \in B$, и для каждого $b \in B$, $n_2(b){\downarrow}$ и $n_2(b) \in A$.

Морфизм $\top : \mathbf{1} \longrightarrow \Omega$ представляется функциональным отношением $F_\top : \{*\} \times P(\mathbb{N}) \longrightarrow P(\mathbb{N})$ вида

$$F_\top(*, A) \longrightarrow (A \Leftrightarrow \mathbb{N}).$$

5.6. Объект натуральных чисел

Объект натуральных чисел N есть объект $(\mathbb{N}, R_N)$ с $R_N(n, n') = \{n| \ n = n'\} = \{n\} \cap \{n'\}$.

Пусть $1 \xrightarrow{0} N$ морфизм, определенный функцией $* \longrightarrow 0$ и представленный функциональным отношением $F_0 : \{*\} \times \mathbb{N} \longrightarrow P(\mathbb{N})$ вида

$$F_0(*, n) = \{0\} \cap \{n\}.$$

Пусть $S : N \longrightarrow N$ морфизм, заданный функцией $n \longrightarrow n+1$ и представленный функциональным отношением $F_S : \mathbb{N} \times \mathbb{N} \longrightarrow P(\mathbb{N})$ вида

$$F_S(n, m) = \{n+1\} \cap \{m\}.$$

Тогда диаграмма

$$1 \xrightarrow{0} N \xrightarrow{S} N$$

есть объект натуральных чисел в топосе $\mathcal{E}\!f\!f$.

В эффективном топосе истинен принцип Брауэра:

Предложение 5.1. (Принцип Брауэра). *Каждая функция $N^N \longrightarrow N$ непрерывна, т. е.*

$$\forall \Phi \in N^{(N^N)} \forall f \in N^N \exists n \forall g \in N^N [\forall i \leq n(f(i) = g(i)) \Rightarrow$$
$$\Rightarrow (\Phi(f) = \Phi(g))]. \qquad (5.1)$$

Доказательство. См.: [87, p. 63]. ■

Таким образом, все функции, определенные на бесконечных словах, вычислимы в топосе $\mathcal{E}\!f\!f$, поскольку в этом случае непрерывность означает вычислимость (см. § 1.5.2, второй подход).

Замечание 5.1. Понять, почему (5.1) означает непрерывность функции Φ, легко, если вспомнить о науке «Топология». Множество функций $\mathbb{N}^{\mathbb{N}}$ рассматриваем как бесконечное произведение $\prod_{n=1}^{\infty} \mathbb{N} = \mathbb{N} \times \mathbb{N} \times ...$ (если $\alpha \in \mathbb{N}^{\mathbb{N}}$, то фактически α – это бесконечный кортеж $(\alpha(0), \alpha(1), ...) \in \mathbb{N} \times \mathbb{N} \times ...$). Снабжаем $\prod_{n=1}^{\infty} \mathbb{N}$ топологией произведения, где $\mathbb{N}$ рассматриваем с дискретной топологией. База данной топологии произведения состоит из множеств U_σ, где $\sigma = (\sigma_0, ..., \sigma_{l(\sigma)-1}))$ – конечная последовательность натуральных чисел длины $l(\sigma)$, и

$$U_\sigma = \{\alpha \in \mathbb{N}^{\mathbb{N}} | \forall n < l(\sigma)[\alpha(n) = \sigma(n)]\}.$$

Тогда с учетом так введенной топологии частичная функция $\Phi : \mathbb{N}^{\mathbb{N}} \longrightarrow \mathbb{N}$ непрерывна в области ее определения $dom(F)$, когда для каждой $\alpha \in dom(F)$ найдется такое $n \in \mathbb{N}$, что для любого $\beta \in dom(F)$,

$$\text{если для всех } i < n, \quad \alpha(i) = \beta(i), \text{ то } \Phi(\alpha) = \Phi(\beta).$$

Получаем равенство (5.1). В принципе, не каждая функция $\Phi : \mathbb{N}^{\mathbb{N}} \longrightarrow \mathbb{N}$ непрерывна, но обязательно все морфизмы $\Phi : N^N \longrightarrow N$ непрерывны в эффективном топосе, как утверждает предложение 5.1.

5.7. Объект действительных чисел

Объект действительных чисел R задаем как объект $R_{rec} = (\mathbb{R}_{rec}, U(\cdot,\cdot))$, где $\mathbb{R}_{rec}$ – множество рекурсивных действительных чисел, т. е. действительных чисел r таких, что существует рекурсивная последовательность Коши, сходящаяся к r [3]. При этом

$$n \in U(r, r')$$

тогда и только тогда, когда $r = r'$ и n кодирует последовательность Коши, сходящуюся к r.

Объект действительных чисел в $\mathcal{E}\!f\!f$ можно строить и как объект R_d, используя понятие сечения Дедекинда, или как R_c, т. е. с помощью последовательностей Коши. Важно, что все три объекта R_{rec}, R_d и R_c изоморфны (подробнее см.: [105]).

Справедлива теорема, утверждение которой для интуиционистской математики предсказывал Брауэр:

---

[3] Назовем последовательность рациональных чисел $\{a_1, ..., a_n, ...\}$ *рекурсивной*, если существуют такие общерекурсивные функции $f(n), g(n), h(n)$ $(h(n) \geq 1)$, что $a_n = (f(n) - g(n)/h(n)$. Будем говорить, что эта последовательность *сходится рекурсивно* (или эффективно), если существует общерекурсивная функция $r(k)$ такая, что для любого как угодно большого $k > 0$

$$|a_n - a_m| < 1/k, \quad \text{если} \quad n, m \geq r(k).$$

Число r, которое определяет эта рекурсивно (эффективно) сходящаяся последовательность, называется *рекурсивным действительным числом* [1].

Теорема 5.1. (Теорема Брауэра в $\mathcal{E}\!f\!f$). *В $\mathcal{E}\!f\!f$ справедливо высказывание «каждая функция из R в R непрерывна»* [60].

Доказательство. См.: [105, p. 160]. ■

5.8. Интерпретация конструктивных логики и арифметики с помощью эффективного топоса

Пусть **L** формальный язык для конструктивных логики и арифметики. Интерпретируем язык **L** в топосе $\mathcal{E}\!f\!f$, выбирая для этого структуру $\mathcal{I} =< N, \cdot, \cdot >$, основой которой является объект натуральных чисел $N = (\mathbb{N}, R_N)$ (см. § 4.6). При такой интерпретации каждой k-местной функции $f(\cdot, ..., \cdot)$ ставится в соответствие морфизм $\bar{f} : (\mathbb{N}^k, .) \longrightarrow (\mathbb{N}, .)$, определенный следующим образом:

$$F_f(\vec{n}, m) = E(\vec{n}) \wedge R_N(f(\vec{n}), m),$$

где $\vec{n} \in \mathbb{N}^k$ [107, p. 21].

Более подробно интерпретация языка **L** в $\mathcal{E}\!f\!f$ дана в [105].

5.9. Вычислимость в топосе $\mathcal{E}\!f\!f$

В синтетическом мире все функции, определенные на натуральных числах, должны быть вычислимыми.

В топосе $\mathcal{E}\!f\!f$ все такие функции, а точнее, все морфизмы $f : N \longrightarrow N$ вычислимы. Это следует из того, что в $\mathcal{E}\!f\!f$ **справедлив конструктивный аналог тезиса Чёрча**, согласно которому *все* функции $f : N \longrightarrow N$ допускают представление (3.1), справедливое для частично рекурсивных функций:

Теорема 5.1. *Имеет место утверждение*

$$\forall f \in N^N \exists e \forall n \exists m (T(e, n, m) \wedge U(m) = f(n)).$$

Доказательство. См.: [87, p. 63], [105, p. 124]. ■

Таким образом, топос $\mathcal{E}\!f\!f$ является синтетическим миром, реализующим условия обязательной вычислимости всех функций.

Справедливость тезиса Чёрча в эффективном топосе означает невозможность в эффективном топосе модели нестандартной арифметики в силу того, что имеет место

Теорема 5.2 (McCarty). *Тезис Чёрча влечет отсутствие нестандартной модели арифметики.* ■

Нестандартная модель арифметики характерна для так называемого *топоса Хербранда* [103], для которого справедив

Ограниченный тезис Чёрча [103]. *Если* $(\forall x \in N)(\exists y \in N)\phi(x,y)$, *то существует вычислимая функция* $f : N \longrightarrow N$ *такая, что*

$$(\forall x \in N)(\exists y \leq f(x))\phi(x,y).$$

5.10. Дж.М. Хайлэнд



Дж. Хайлэнд

Джон Мартин Эллиот Хайлэнд (Hyland), профессор математики Королевского колледжа в Кэмбриджском университете (Англия).

Докторская диссертация (Ph.D.) на тему «Recursion Theory on the Countable Functionals» (1975) была написана и защищена под руководством Робина Гэнди (Robin Gandy) – друга и ученика Алана Тьюринга.

Хайлэнд работает в областях математической логики, теории рекурсивных функций, теории категорий и теоретических компьютерных наук.

В 1982 году открыл эффективный топос, который используется в компьютерных науках как инструмент порождения моделей языков программирования (например, the Calculus of Constructions).

Глава 6

Топосы реализуемости над частичными комбинаторными алгебрами

Синтетический мир, открывающийся в эффективном топосе, строится сразу как топос. Этот способ предоставляет в распоряжение исследователя структуру, которую можно использовать для интерпретации формальных интуиционистских теорий высокого порядка.

Имеется другой способ конструирования эффективного топоса, с помощью которого получается эффективный топос как частный случай. Этот способ основывается на построении категорий, использующих *частичные комбинаторные алгебры* (pca), или *алгебры Шейнфинкеля*. С помощью такой алгебры A строится категория $Ass(A)$ ассамблей и их морфизмов и производится ее пополнение до категории, являющейся *топосом реализуемости* $RT(A)$. Одним из топосов реализуемости является эффективный топос.

Первоначально топосы реализуемости были получены с помощью теории трипосов [72], [93], но затем Freyd вместе с коллегами

[61] показал, что лучше начинать с категории ассамблей $Ass(A)$ и пополнять ее.

В гл. 5 было рассказано об интерпретации конструктивных логики и арифметики с помощью эффективного топоса. В качестве номеров рекурсивных функций, используемых для реализации формул, рассматривались натуральные числа. Возможно обобщение этого подхода, когда в качестве номеров берутся элементы так называемых *частичных комбинаторных алгебр* A.

Реализуемость Клини является частным случаем использования частичных комбинаторных алгебр, поскольку представляется в теории pca первой алгеброй Клини K_1. Каждая pca обладает множеством натуральных чисел и обычные числовые частично рекурсивные функции $f : \mathbb{N} \longrightarrow \mathbb{N}$ имеют представления во всех частичных комбинаторных алгебрах. Частичные комбинаторные алгебры предоставляют множество возможностей различной интерпретации конструктивных доказательств. При этом топосы реализуемости дают интерпретации их высокопорядковых логик и помогают понять эти логики [67].

Топосы реализуемости являются обобщением идеи реализуемости Клини, представляющей собой интерпретацию конструктивистской (интуиционистской) арифметики (и теории чисел) как части их внутренней логики.

Топосы реализуемости не являются топосом Гротендика и даже не имеют геометрического морфизма в категорию **Sets**.

Реализуемость, базирующаяся на *частичных комбинаторных алгебрах* (pca), называется *стандартной.*

Частичные комбинаторные алгебры обобщают формулу

$$\{m\}(n) = \textbf{образ рекурсивной функции с номером } m \textbf{ числа } n,$$

которая является по сути дела применением натурального числа к натуральному числу путем рассмотрения *операции применения* $A \times A \longrightarrow A$ к элементам множества A, обладающей свойствами:

$$(ka)b = a, \quad ((sa)b)c \asymp (ac)(bc),$$

которые воспринимаются как непонятные искусственно придуманные равенства. В действительности они являются переносом на pca равенств, установленных для частично рекурсивных функций в $s - m - n$-теореме [24, с. 88].

Частичные кобинаторные алгебры – это модели абстрактной теории вычислимости. С различными pca появляются различные

понятия вычислимости. Насколько эти понятия вычислимости существенны и значимы, проясняется по мере построения различных моделей реализуемости [82, p. 267].

Таким образом, топосы реализуемости – это «модели конструктивистской теории множеств», основанные на абстрактной теории вычислимости [82, p. 1].

6.1. Алгебры Шейнфинкеля

Определение 6.1. *Частичная комбинаторная алгебра* (pca), или *алгебра Шейнфинкеля*, – это множество A с частичной *операцией применения* $-\cdot- : A\times A \longrightarrow A$ и двумя выделенными элементами $k, s \in A$, для которых справедливы соотношения

$$(ka)b \simeq a, \quad (sa)b\downarrow, \quad ((sa)b)c \simeq (ac)(bc), \tag{6.1}$$

где $\simeq$ означает, что выражения в обеих сторонах определены и равны, а $\downarrow$ говорит, что выражение, левее этого знака, определено. Пишем xy вместо $x\cdot y$, и элемент слева применяется к элементу справа; pca *нетривиальная*, если $k \neq s$.

Пример 6.1. (**Первая алгебра Клини** K_1). Это множество натуральных чисел $\mathbb{N}$, снабженное операцией применения Клини $n\cdot m = \{n\}m$, которая применяет частично рекурсивную функцию $\{n\}$ с гёделевским номером n к числу m. Существование элементов k, s со свойствами (6.1) следует из так называемой $s{-}m{-}n$-теоремы [24, с. 88]. Собственно, эта теорема и взята в качестве условий (6.1), выделивших особый статус pca.

Пример 6.2. (Engeler's Graph Algebra). Пусть A непустое множество. Полагаем

$G_0(A) = A,$

$$G_{n+1}(A) = G_n(A) \cup \{(B,b) | B \subseteq G_n(A), B \text{ конечно}, b \in G_n(A)\}.$$

Пусть

$$G(A) = \bigcup_{n\in\mathbb{N}} G_n(A).$$

Тогда $P(G(A))$ есть pca, если принять, что

$$X\cdot Y = \{b \in G(A) | \exists B \subseteq Y : (B,b) \in X\}.$$

Предложение 6.1. *Существуют* $p, p_0, p_1 \in A$ *такие, что*

$$pab\downarrow,\ p_0(pab) = a,\ p_1(pab) = b$$

для всех $a, b \in A$.

Существуют $\mathrm{true}, \mathrm{false}, \mathrm{cond} \in A$ *такие, что*

$$\mathrm{cond}\ ab\downarrow,\ \mathrm{cond}\ ab\ \mathrm{true} = a,\ \mathrm{cond}\ ab\ \mathrm{false} = b$$

для всех $a, b \in A$ [101, p. 19]. ■

Вместо pab пишем $\langle a, b\rangle$. Это пара из элементов a и b.

Вводим обозначение: $i \stackrel{def}{=} skk$. Тогда $ix = x$ для любого $x \in A$.

6.1.1. Нумералы и рекурсия в pca

Определение 6.2 (Нумералы). Каждому натуральному числу n поставим в соответствие элемент $\overline{n} \in A$. Полагаем по индукции:

$$\overline{0} = \langle \mathrm{true}, i\rangle, \quad \overline{n+1} = \langle \mathrm{false}, \overline{n}\rangle.$$

Элемент $\overline{n}$ называется *нумералом* числа n [101, p. 19].

Таким образом, множество натуральных чисел $\mathbb{N}$ вкладывается в pca A.

Предложение 6.2. *Существуют* $\mathrm{succ}, \mathrm{pred}, \mathrm{isz} \in A$ *такие, что*

$$\mathrm{succ}\ \overline{n} = \overline{n+1},\ \mathrm{pred}\ \overline{0} = \overline{0},\ \mathrm{pred}\ \overline{n+1} = \overline{n},$$

$$\mathrm{isz}\ \overline{0} = \mathrm{true},\ \mathrm{isz}\ \overline{n+1} = \mathrm{false}$$

для всех $n \in \mathbb{N}$. ■

Теорема 6.1 (Примитивная рекурсия)**.** *Существует* $\mathrm{rec} \in A$ *такой, что*

$$\begin{aligned} &\mathrm{rec}\ a\ f\ \overline{0} = a,\\ &\mathrm{rec}\ a\ f\ \overline{n+1} \simeq f\ \overline{n}(\mathrm{rec}\ a\ f\ \overline{n}) \end{aligned} \tag{6.2}$$

для всех $a, f \in A$ *и* $n \in \mathbb{N}$. ■

Благодаря константе rec строится абстрактная теория примитивно вычислимых функций над pca. А также строится абстрактная теория частично рекурсивных функций. Тем самым с помощью pca создается абстрактная теория вычислимости.

Теорема 6.2 [55]. *Пусть дана pca A и $f : \mathbb{N}^k \longrightarrow \mathbb{N}$ – примитивно рекурсивная функция. Тогда существует $e \in A$ такой, что для любых $m_1, ..., m_k, n \in \mathbb{N}$ имеем*

$$f(m_1, ..., m_k) = n \Rightarrow e\overline{m}_1...\overline{m}_k = \overline{n}.$$

■

Как видим, любая примитивно рекурсивная функция *представляется* в частичных комбинаторных алгебрах.

6.1.2. Морфизмы pca

Определение 6.3. Пусть A и B две pca. *Применяющий морфизм* $A \longrightarrow B$ – это отображение $\gamma : A \longrightarrow P(B)$ из A в множество непустых подмножеств множества B, которое имеет *реализатор*, т. е. элемент $r \in B$, удовлетворяющий следующему условию: для любых $a, a' \in A$ таких, что если $aa' \downarrow$ и $b \in \gamma(a)$, $b' \in \gamma(a')$, то $rbb' \downarrow$ и $rbb' \in \gamma(aa')$.

6.2. Категория ассамблей $Ass(A)$

Пусть A – частичная комбинаторная алгебра.

Ассамблея X над A есть множество $|X|$ вместе с функцией, которая каждому элементу $x \in |X|$ ставит в соответствие непустое подмножество $||x||_X \in P(A)$.

Пишем $X = (|X|, || \cdot ||_X)$.

Множество $|X|$ называем базой для X, а функцию $|X| \ni x \longrightarrow ||x||_X$ – *реализатором* для x и если $a \in ||x||_X$, то говорим, что a *реализует* x.

Пусть X, Y две ассамблеи. Говорим, что функция $f : |X| \longrightarrow |Y|$ *отслеживается* элементом $r \in A$, если для любых $x \in |X|, a \in A$ имеем

$$a \in ||x||_X \Rightarrow ra \in ||f(x)||_Y.$$

Морфизм $F : X \longrightarrow Y$ есть функция $f : |X| \longrightarrow |Y|$, отслеживаемая некоторым $r \in A$.

Определение 6.2. Ассамблеи и морфизмы между ними образуют декартово замкнутую *категорию ассамблей* $Ass(A)$.

Интуитивная интерпретация категории $Ass(A)$. Ассамблею X можно воспринимать как *тип данных*[1], а $|X|$ как множество значений, относящихся к данному типу данных. Подмножество $||x||_X \in P(A)$ состоит из *машинных реализаций* значения $x \in |X|$. Морфизм $f : X \longrightarrow Y$ между двумя типами данных – это функция, переводящая значения одного типа данных в значение другого типа данных. Если r отслеживает f, то это понимается как автоматическое (машинное) преобразование типов данных, когда мы присваиваем одной переменной значение переменной другого типа.

Данная интерпретация ассамблей является примером методов *денотационной семантики.*

6.3. Денотационная семантика

Денотационная семантика (англ. denotational semantics) имеет своей задачей поиск всевозможных строгих описаний языков программирования, давая различные интерпретации программам в терминах и понятиях используемых для этого математических структур. Денотационная семантика выражениям в программе ставит в соответствие настоящие математические объекты используемых для этого математических структур, то есть выражения обозначают (англ. to denote – «обозначить»; откуда «денотационная») их величины [45, с. 593–594].

Самой простой интерпретацией является понимание программы, которая, имея на входе одни данные (одного типа данных), переводит их на выходе в другие данные (другого типа данных), как функции (морфизмы в общем случае), преобразующей входные данные в выходные данные. Поскольку теория категорий и занимается преобразованием одних объектов в другие, то первое, что приходит в голову, – использовать теорию категорий для решения проблем, возникающих в программировании. Кстати, идея использования натуральных чисел в качестве элементов типов данных принадлежит Ю.Л. Ершову [21], [64], который рассмотрел категорию (тотальных) отношений эквивалентности на $\mathbb{N}$.

Что дает денотационная семантика программистам? Программисты, когда пишут программы, подразумевают, что код, который они создали, будет выполнять определенные действия, приводящие

[1] Под типом данных в программировании понимают множество объектов одного и того же сорта.

к желаемым результатам. И, как правило, очень удивлены, если это не так. Почему такое случается? Просто по той причине, что программист, созидая код, неосознанно прогоняет его в своей голове. Но в программе может быть слишком много параметров, не за всеми уследишь – голова не компьютер, не все может проверить заранее. Поэтому желательно иметь возможность до запуска программы проверить те или иные ее свойства, ее параметры, на корректность исполнения. А математика как раз и создана человечеством для доказательств того, что нечто обладает нужным свойством. Поэтому естественно использовать методы математических доказательств для проверки наличия требуемых свойств у программного кода, проинтерпретировав программу в терминах специально подобранной для этого математической структуры. Иначе говоря, «если вы хотите доказать свойство программы, вы просто доказываете математическую теорему». Программист может возразить, «что доказывание теорем трудно, но на самом деле мы, люди, строили математические методы тысячи лет, так что есть множество накопленных знаний, которые можно использовать. Кроме того, по сравнению с теоремами, которые доказывают профессиональные математики, задачи, с которыми мы сталкиваемся в программировании, как правило, довольно просты, если не тривиальны»[2].

Одним из возможных направлений денотационной семантики является привлечение в качестве интерпретаций математических структур, именуемых *топосами реализуемости*. Появившись в целях конструктивных интерпретаций интуиционистской логики и арифметики, топосы реализуемости в последнее время используются в компьютерных науках.

6.4. Топос реализуемости $RT(A)$

Категория $Ass(A)$ не является точной. Применим к ней операцию точного пополнения, изложенную в § 4.10.3. Справедлива следующая

Теорема 6.3. *Категория $EC(Ass(A))$ есть топос.*

Доказательство. См.: [69]. ■

[2] *Milewski B.* Category Theory for Programmers: The Preface. URL: http://bartoszmilewski.com/2014/10/28/category-theory-for-programmers-the-preface/.

6.4.1. Стандартные топосы реализуемости

Определение 6.4. Категория $EC(Ass(A))$ называется (стандартным) *топосом реализуемости* и обозначается как $RT(A)$.

Имеем

$$\boxed{i_{EC} : Ass(A) \longrightarrow RT(A) \equiv EC(Ass(A)).} \tag{6.3}$$

Таким образом, полностью удается заменить множество натуральных чисел $\mathbb{N}$, используемое для нумерации алгоритмов (рекурсивных функций), на частичные комбинаторные алгебры A и получить при этом более общую топосную структуру для конструктивной интерпретации – топос реализуемости, частным примером которого является эффективный топос.

Топосы реализуемости – это «модели конструктивной теории множеств», основанные на абстрактном понятии вычислимости [51]:

- Каждый топос $RT(A)$ имеет объект натуральных чисел.
- Топос реализуемости не является топосом Гротендика.

6.4.2. Топос $\mathcal{E}\!f\!f$ как топос реализуемости

В случае первой алгебры Клини

$$\boxed{RT(K_1) = \mathcal{E}\!f\!f,}$$

т. е. топос реализуемости $RT(K_1)$ – это эффективный топос, дающий нам математический мир с синтетической вычислимостью.

Ответ на вопрос, являют ли топосы реализуемости $RT(A)$ синтетический мир, в котором все морфизмы $f : N^k \longrightarrow N$, где N – объект натуральных чисел в $RT(A)$, – это вычислимые функции, нам пока не известен.

Представим два результата, которые демонстрируют определенные отличия некоторых топосов реализуемости от эффективного топоса.

Говорим, что pca A является *тотальной*, если для любых $a, b \in A$ $ab\downarrow$.

Справедливы утверждения:

- Эффективный топос $RT(K_1)$ не эквивалентен $RT(A)$ над тотальной A [67].
- Каждый топос реализуемости есть геометрическая сюрьекция (quotient) топоса реализуемости над тотальной pca [104].
- Существует топос реализуемости $RT(J)$, в котором существует инъекция $N^N \longrightarrow N$ [52].

6.5. М.Э. Шейнфинкель

Моисей Эльевич Шейнфинкель (Schönfinkel, 4 сентября 1889, Екатеринослав (Днепропетровск) – 1942, Москва) – российский логик и математик. Закончил Новороссийский университет (Одесса). Изучал математику под руководством Самуила Осиповича Шатуновского (1859–1929), работавшего в области геометрии и оснований математики.



М.И. Шейнфинкель (1910)

С 1914 по 1924 год Шейнфинкель был членом группы Гильберта в Геттингенском университете (Германия). Перед Математическим обществом Геттингена 7 декабря 1920 года сделал доклад, который принес ему известность. В докладе он изложил концепцию комбинаторной логики.

Давид Гильберт рекомендовал доклад к публикации, которая и была осуществлена четырьмя годами позже в математическом журнале Mathematische Annalen [98].

Шейнфинкелем была разработана формальная система, позволяющая избегать использования связанных переменных. Его система была по существу эквивалентна комбинаторной логике, основанной на комбинаторах b, c, i, k и s. Шейнфинкель показал, что система может быть сокращена до k и s.

Шейнфинкель предложил простую теорию функций, которая фактически являлась исчислением объектов-функций и предвосхитила появление лямбда-исчисления – математической формали-



БИЛЕТЪ

для постояннаго ношенія при себѣ и предъявленія при входѣ въ зданіе Университета.

№ [illegible]

Шейнфинкель

Моисей

Эльевичъ

Осень 1909/10 уч. года.

И. д. Проректора

Секретарь по студ. дѣламъ

СТУДЕНТЪ

ИМПЕРАТОРСКАГО

Новороссійскаго Университета

Физико-Математическаго

ФАКУЛЬТЕТА

Матем. отд.

Весна 1909/10 учеб. года.

Проректоръ

Секретарь по студ. дѣламъ

Рис. 6.1. Студенческий билет М.Э. Шейнфинкеля

зации, поддерживающей языки функционального программирования. Было показано, что функции двух или более аргументов могут быть заменены функциями, принимающими лишь один аргумент. Механизм такой замены упрощает работу как в терминах комбинаторной логики, так и лямбда-исчисления, и позднее был назван *каррированием*, в честь Хаскелла Карри.

Основная работа – «О строительных блоках математической логики» [97].

В 1929 году им была подготовлена совместная с Бернайсом публикация [57], посвященная проблеме разрешимости для одного частного класса формул исчисления предикатов.

В конце 1920-х вернулся в СССР. Последние годы жизни провел в Москве, где страдал от нищеты и психического расстройства. Умер в 1942 году. Во время войны бывшие соседи Шейнфинкеля сожгли все его рукописи в печи [48].

Глава 7

Топосы пучков над алгебрами Шейнфинкеля

Алгебры Шейнфинкеля, или *частичные комбинаторные алгебры* (pca) A, используются при построении различных топосов, которые можно рассматривать как математико-теоретический аппарат и для создания мира синтетической вычислимости по Банаху–Мазуру в форме топоса пучков над pca A и для решения проблемы трансляции различных типов данных языков программирования.

7.1. Категория умеренных множеств

7.1.1. Категория $Mod(A)$

Пусть A частичная комбинаторная алгебра (pca).

Умеренное множество $(S, \Vdash_S)$ над A есть множество S с *отношением реализуемости* $\Vdash_S \subseteq A \times S$ таким, что для каждого $x \in S$ существует $a \in A$, для которого $a \Vdash_S x$, и для всех $x, y \in S$,

$$((a \Vdash_S x) \wedge (a \Vdash_S y)) \Leftrightarrow x = y.$$

Реализованная функция $f : (S, \Vdash_S) \longrightarrow (T, \Vdash_T)$ между умеренными множествами есть функция $f : S \longrightarrow T$, которая *отслеживается* некоторым $a \in A$, т. е. для любых $x \in S$, $b \in A$

$$(b \Vdash_S x) \Rightarrow ab \downarrow \wedge (ab \vDash_T f(x)).$$

Напомним, что знак $\downarrow$ говорит о том, что выражение, левее этого знака, определено.

Определение 7.1. Категорию умеренных множеств и реализованных функций обозначаем как $Mod(A)$.

Замечание 7.1. Категория $Mod(K_1)$ тесно связана с категорией нумерованных множеств Ершова En. *Нумерованные множества Ершова* – это (везде определенные) сюрьекции из $\mathbb{N}$ (см. § 8.1.1). Умеренные множества над K_1 эквивалентны категории частичных нумераций, которые являются частичными сюрьекциями $\mathbb{N} \twoheadrightarrow S$.

Категория $Mod(A)$ может быть введена иначе, поскольку возможно определение умеренного множества над A как частного случая ассамблеи [82, p. 39].

Ассамблея $X = (|X|, || \cdot ||_X)$ над A называется *умеренным множеством над* A, если для любых $x, y \in |X|$

$$x \neq y \Leftrightarrow ||x||_X \cap ||y||_X = \emptyset. \tag{7.1}$$

При принятии такого определения умеренных множеств легко устанавливается, что категория $Mod(A)$ умеренных множеств A является полной подкатегорией категории ассамблей $Ass(A)$ и имеет место включение

$$\boxed{j : Mod(A) \hookrightarrow Ass(A).} \tag{7.2}$$

Разница между категориями $Ass(A)$ и $Mod(A)$ с точки зрения интерпретации элементов x как значения с типом данных X, а элементов $a \in A$ как машинных реализаций значений типа данных в том, что в $Ass(A)$ одна и та же машинная реализация a может представлять более чем одно значение (x и y, $x \neq y$) типа данных X [82, p. 16]. Это видно из формулы (7.1).

Категории $Mod(A)$ и $Ass(A)$ имеют объект натуральных чисел [82, p. 40].

7.1.2. Категория $\langle A \rangle$

Пусть A частичная комбинаторная алгебра.

Объекты категории $\langle A \rangle$ – это подмножества $X \in P(A)$, а *морфизмы* $f : X \longrightarrow Y$ – реализованные отображения между подмножествами pca, т. е. функция $f : X \longrightarrow Y$ – морфизм, если для нее существует $a \in A$ такой, что для всех $x \in X$, $ax \downarrow$ и $f(x) = ax$.

Каждый объект $X \in \langle A \rangle$ является умеренным множеством $(X, || \cdot ||_X)$. *Реализатор* $x \longrightarrow ||x||_X$ определяется как

$$||x||_X = \{x\} \subset P(A).$$

При этом морфизм между $\langle A \rangle$-объектами оказывается, как легко проверить, морфизмом между ними как ассамблеями. Следовательно, имеет место включение

$$\boxed{i : \langle A \rangle \hookrightarrow Mod(A).} \tag{7.3}$$

7.2. Топосы пучков над pca A

В качестве топологии Гротендика на $\langle A \rangle$ рассматриваем топологию копроизведения J, которая порождается такими семействами $(f_i : Y_i \longrightarrow X)_{i \in I}$, что копроизведения $\amalg_{i \in I} Y_i$ существуют[1] и морфизм $[f_i]_{i \in I} : \amalg_{i \in I} Y_i \longrightarrow X$ есть изоморфизм.

Определение 7.1. Топос пучков на $\langle A \rangle$ с топологией копроизведения C обозначим как $Sh_C(A)$.

Теорема 7.1. *Категория $Sh_C(A)$ эквивалентна категории пучков $Sh_G(Mod(A))$ с подканонической топологией Гротендика G, порождаемой такими семействами $(f_i : Y_i \longrightarrow X)_{i \in I}$, для которых $\amalg_{i \in I} Y_i$ существует и морфизм $[f_i]_{i \in I} : \amalg_{i \in I} Y_i \longrightarrow X$ есть регулярный эпиморфизм* [51, p. 6–7]. ■

[1] Доказано, что такие семейства имеются в $\langle A \rangle$ при $|I| < w(A)$, где $w(A)$ – некоторый кардинал [51, p. 5].

Теорема 7.2. *Имеет место вложение Ионеды*

$$\boxed{y : \langle A\rangle \hookrightarrow Sh_C(A)} \tag{7.4}$$

и вложение

$$\boxed{\widetilde{y} : Mod(A) \hookrightarrow Sh_C(A),} \tag{7.5}$$

$$\widetilde{y} = y \circ i,$$

которое является полным и точным. ■

Строгое определение вложения (7.5) подразумевает, что каждому $S \in Mod(A)$

$$\widetilde{y}(S)(-) = Hom_{Mod(A)}(i(-), S),$$

где

$$i : \langle A\rangle \hookrightarrow Mod(A)$$

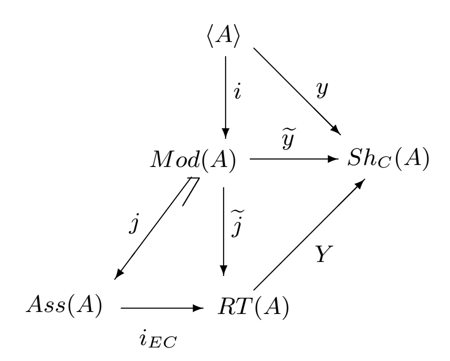
– включение.

7.3. Диаграмма для категорий $\langle A\rangle$, $Mod(A)$, $Ass(A)$, $RT(A)$ и $Sh_C(A)$

В этой главе на первом этапе мы построили категорию $Mod(A)$ умеренных множеств и реализованных морфизмов (функций) и категорию $\langle A\rangle$. Затем категорию $\langle A\rangle$ оснастили топологией Гротендика C и преобразовали в топос пучков $Sh_C(A)$.

Категория $Mod(A)$, как было отмечено, является полной подкатегорией категории $Ass(A)$, т. е. имеется вложение $Mod(A) \hookrightarrow Ass(A)$.

Все построенные категории находятся в отношениях, которые фиксируются в форме функторов между ними, представленных формулами (6.3), (7.2)–(7.5). Объединяя их, получаем диаграмму:



Здесь $i : \langle A \rangle \longrightarrow Mod(A)$ включение. Для $S \in Mod(A)$, $\widetilde{y}(S)(-) = Hom_{Mod(A)}(i(-), S)$. Наконец, $\widetilde{j} = i_{EC} \circ j : Mod(A) \longrightarrow RT(A)$ – включение и функтор Y определяется подобно вложению Ионеды y, т. е. если $X \in RT(A)$, то $Y(X)(-) = Hom_{RT(A)}(j(-), X)$.

7.4. Категория частичных отношений эквивалентности $PER(A)$

Определение 7.2 (PERs).

(i) Пусть дана pca A. *Частное отношение эквивалентности* (или PER) на A есть симметричное, транзитивное бинарное отношение R на A. Пишем A/R для множества классов эквивалентности по отношению R.

(ii) Если R, S являются PER на A, то PER S^R определяется следующим образом:

$$S^R(a, a') \Leftrightarrow (\forall b, b' \in A)(R(b, b') \Rightarrow S(ab, a'b')).$$

Элемент множества $A/(S^R)$ рассматриваем как морфизм $f : R \longrightarrow S$. Говорим, что *a реализует* f, если $a \in f$.

Пишем $PER(A)$ для *категории всех* PER *на* A *и морфизмов между ними.*

Справедливы утверждения:

$$PER(K_1) \text{ подкатегория } \mathcal{E}\!f\!f,$$

$$PER(A) \text{ подкатегория } RT(A).$$

Теорема 7.3. *Категории $Mod(A)$ и $PER(A)$ эквивалентны.*

Доказательство. См.: [54, p. 37]. ∎

Элементы типа данных в компьютере представляются их машинными реализациями – конечной последовательностью битов, т. е. цифровым кодом. Два разных элемента типа данных могут иметь один и тот же код. Переходя к абстрактной математической теории, цифровой код лучше заменить натуральными числами. В таком случае переход от одного типа данных к другому можно описывать как вычисление (компьютер ведь вычисляет), и использовать для этого рекурсивные функции. При еще более высоком уровне абстракции натуральные числа меняются на элементы алгебры Шейнфинкеля A и используются представления рекурсивных функций в A (см. теорему 6.2). Вспоминая, что один код может задавать разные элементы типа данных, «склеиваем» эти данные посредством отношения эквивалентности на A. Таким образом, тип данных характеризуется некоторым отношением эквивалентности, т. е. объектом категории $PER(A)$.

7.5. Реализуемость и модели вычислимости

Можно расширить понимание реализуемости, под которой до сих пор имелась в виду вычислимость истинности математических и логических утверждений с помощью классических рекурсивных функций или с помощью рекурсивных функций, представленных в частичных рекурсивных алгебрах, до понимания под реализуемостью погружения любых математических и логических структур S в любые возможные *модели вычислимости* $\mathcal{M}$, с помощью которых будет вычисляться истинность утверждений, формулируемых в рамках этих структур.

Под моделями вычислимости $\mathcal{M}$, помимо рекурсивной вычислимости, можно понимать, например, вычислимость с помощью машин Тьюринга, вычислимость с помощью набора корректных программ в том или ином языке программирования, с помощью хорошо сформированных блок-схем, описывающих алгоритмы, и т. д. [54, p. 11].

Модель вычислимости не только определяет множества допустимых операций, используемых для вычисления, но также задает ограничения, накладываемые на вычисления, и относительные издержки, неизбежно сопровождающие выбор той или иной модели вычислимости. Охарактеризовать необходимые вычислительные ресурсы – время выполнения, объём памяти, а также ограничения алгоритмов или компьютера – можно только в том случае, если выбрана конкретная модель вычислимости.

Под погружением структуры S в модель вычислимости $\mathcal{M}$, или под *реализацией*, понимается сопоставление каждому элементу $s \in S$ программы (вычисления) $p \in \mathcal{M}$. Появляется *отношение реализуемости* $\Vdash_S$, или $p \Vdash_S s$ (p реализует $s \in S$).

Отношение реализуемости должно удовлетворять двум условиям:

1) каждый $s \in S$ имеет хотя бы один реализатор p;

2) каждая программа p реализует хотя бы один элемент $s \in S$ [2].

Множество S с отношением реализуемости $(S, \Vdash_S)$ называем *умеренным множеством*[3] (modest set).

Если $(S, \Vdash_S)$ и $(T, \Vdash_T)$ умеренные множества, то функция $f : S \longrightarrow T$ называется *реализованной* программой $p \in \mathcal{M}$, когда $a \vDash_S x$ влечет $p(a) \Vdash_S f(x)$. Другими словами, p отображает реализаторы для x в реализаторы для $f(x)$. (Элементы $\mathcal{M}$ могут быть как программами, так и данными; не будем делать различия между ними.)

Говорят также, что $p \in \mathcal{M}$ *отслеживает* f, и пишут $p \Vdash_{S \longrightarrow T} f$. Функция между умеренными множествами, которая имеет реализатор, называется *реализованной* функцией.

Таким образом, мы расширили понятие реализуемости и построили расширенную категорию умеренных множеств.

Моделью вычислимости может служить использование монад из теории категорий. Еугенио Моджи обнаружил, что вычислительные эффекты могут быть преобразованы в монады. Это оказалось важным наблюдением, которое не только дало денотационной семантике новую жизнь и сделало чисто функциональные программы более удобными, но и дало новую информацию о традиционном программировании (Milewski)

---

[2] Второе условие не является необходимым, и ведет к категории ассамблей.

[3] Умеренное, поскольку мощность множества S не должна быть больше мощности $\mathcal{M}$.

Глава 8

Рекурсивный топос

Рекурсивный топос Малри RtM – это математический мир, подходящий для теории синтетической вычислимости в том случае, когда под вычислимыми функциями понимаются функционалы $f : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ (функции, определенные на бесконечных словах). Другими словами, рекурсивный топос моделирует вычислимость по Банаху–Мазуру.

8.1. Категория нумерованных множеств Ершова

8.1.1. Нумерации множества и его подмножеств

Пусть S – произвольное непустое не более чем счетное множество. *Нумерацией* множества S назовем всякую сюрьекцию (т. е. отображение «на») ν множества $\mathbb{N}$ всех натуральных чисел на множество S.

Пара $\Sigma = (S, \nu)$, где ν – некоторая нумерация множества S, называется *нумерованным множеством*[1]. Для дальнейшего нам

[1] Обобщением понятия нумерованного множества являются умеренные множества. Для последних вместо нумерации задается отношение реализуемости.

будет удобно считать, что и пустое множество $\emptyset$ обладает некоторой единственной «нумерацией» o, а «нумерованное» множество $(\emptyset, o)$ будем обозначать $\mathbf{O}$.

Если $\nu(n) = s, s \in S$, то говорят, что n есть код, или рекурсивный индекс (см. § 3.1.2) для s.

8.1.2. Категория Ершова En

Объектами *категории Ершова* En являются все нумерованные множества, включая «пустое» нумерованное множество $\mathbf{O}$. Если Σ – произвольное нумерованное множество, то существует, и притом единственный, морфизм 0 из $\mathbf{O}$ в Σ.

Если $\Sigma_1 = (S_1, \nu_1)$ и $\Sigma_2 = (S_2, \nu_2)$ – произвольные непустые нумерованные множества, то *морфизмом из* Σ_1 *в* Σ_2 назовем всякое отображение $\mu : S_1 \longrightarrow S_2$, для которого существует общерекурсивная функция $f : \mathbb{N} \longrightarrow \mathbb{N}$ такая, что $\mu \circ \nu_1 = \nu_2 \circ f$, иными словами, если диаграмма

$$\begin{array}{ccc} \mathbb{N} & \xrightarrow{f} & \mathbb{N} \\ \nu_1 \downarrow & & \downarrow \nu_2 \\ S_1 & \xrightarrow{\mu} & S_2 \end{array}$$

коммутативна.

Композиция морфизмов определяется естественным образом. Объект $\mathbf{O}$ является нулевым объектом категории En. Категория En обладает объектом натуральных чисел.

Определенные нами морфизмы называют часто *эффективными морфизмами* [88], хотя, казалось бы, следует говорить о вычислимом морфизме. Дело в том, что в математической семантике языков программирования понятие эффективной вычислимости не употребляется, поскольку оно относится к способу организации вычислений, к их сложности, как и само понятие вычислимости связывают с конкретными процедурами, с конкретными вычисляемыми элементами языка программирования.

Понятие эффективности оказывается более удобной заменой понятия вычислимости. Оно подразумевает оперирование (вычисление), производимое с изучаемыми абстрактными типами данных. При необходимости возможно дать определение вычислимых элементов, привязываемых к изучаемой эффективности. Выше это было сделано привязкой морфизмов к рекурсивным функциям и вве-

дением понятия «эффективный морфизм», т. е., по сути, речь идет о вычислимом морфизме.

Категория En – это Мир эффективности [88], т. е. мир синтетической вычислимости. Но он не является топосом, поскольку En далеко не декартово замкнута и отсутствует экспоненцирование. Однако ее можно расширить до декартово замкнутой категории *обобщенных нумерованных множеств* GEn [88]. Морфизмы в GEn подобны функционалам Мазура–Банаха [94], переводящим рекурсивные функции в рекурсивные.

Категория En погружается также в рекурсивный топос Малри RtM, чьи морфизмы суть функционалы Мазура–Банаха, и эквивалентна полной подкатегории топоса RtM.

8.2. Рекурсивный топос Малри

Определим три категории: Rec, PR и $Dom(PR)$.

Категорию Rec называют моноидом одноместных общерекурсивных функций. Она имеет единственный объект – множесто **Rec** всех одноместных общерекурсивных функций. Морфизмы – это элементы $f, g, \ldots$ множества **Rec**, композиция морфизмов $f \circ g$ – композиция функций, единичный морфизм совпадает с тождественной функцией.

Категория PR состоит из одного объекта $\mathbb{N}$ и морфизмов $f : \mathbb{N} \longrightarrow \mathbb{N}$, которыми являются частично рекурсивные функции.

Категория $Dom(PR)$ состоит из объектов $A \subset \mathbb{N}$ и морфизмов вида $f : A \longrightarrow B$, $A, B \subset \mathbb{N}$, где f – частично рекурсивная функция, определенная на всем A.

Определение 8.1. *Рекурсивный топос Малри* [91] – это топос

$$RtM = Sh_J(\mathcal{K}),$$

где:

$\mathcal{K}$ – это либо моноид одноместных общерекурсивных функций Rec, либо $Dom(PR)$, либо En;

J – каноническая топология Гротендика.

Из определения предпучка и вложения Ионеды следует, что

имеют место вложения [95, p. 76]

$$\boxed{Rec,\ Dom(PR),\ En \overset{y}{\hookrightarrow} RtM.} \tag{8.1}$$

Как видим, рекурсивный топос является топосом пучков, следовательно, кополным, тогда как эффективный топос имеет только конечные (нетривиальные) копроизведения.

Вложение

$$Rec \overset{y}{\hookrightarrow} RtM$$

говорит о том, что объекту **Rec** соответствует в рекурсивном топосе функтор $y(\mathbf{Rec})(-) = Hom_{Rec}(-, \mathbf{Rec}) : Rec^{op} \longrightarrow \mathbf{Sets}$, а Rec–морфизму $f \in Hom_{Rec}(\mathbf{Rec}, \mathbf{Rec})$, т. е. рекурсивной функции f отвечает естественное преобразование

$$\tau_{-} = y(f)(-) : Hom_{Rec}(-, \mathbf{Rec}) \longrightarrow Hom_{Rec}(-, \mathbf{Rec}).$$

Поскольку в категории Rec всего один объект **Rec**, то τ имеет только одну компоненту

$$\tau_{\mathbf{Rec}} = y(f)(\mathbf{Rec}) : Hom_{Rec}(\mathbf{Rec}, \mathbf{Rec}) \longrightarrow Hom_{Rec}(\mathbf{Rec}, \mathbf{Rec}).$$

Таким образом, естественные преобразования, отвечающие рекурсивным функциям, – это отображения, переводящие рекурсивные функции в рекурсивные, или вычислимые в вычислимые.

Следовательно, рекурсивный топос моделирует вычислимость по Банаху–Мазуру [84], [38], в то время как эффективный топос моделирует вычислимость по Клини (такие отображения реализуются на машине Тьюринга).

8.3. Категория эффективных доменов Скотта *EDom*

Пусть $D =< D, \preceq >$ – частично упорядоченное множество (сокращенно – «чум»).

Отношение $x \preceq y$ Скотт трактует следующим образом: «информация, содержащаяся в y, включает информацию, содержащуюся в x, и именно потому y лучше x» [40, с. 29–31].

Множество $X \subset D$ называется *направленным*, если $X \neq \emptyset$ и

$$\forall x, y \in X \exists z \in X (x, y \preceq z).$$

Чум D называется *полным чумом*, если:

1) в D есть наименьший элемент, обозначаемый символом $\perp$ и называемый «дно»;

2) для любого направленного $X \subset D$ множество D содержит его точную верхнюю грань (обозначаемую $\sqcup X \in D$).

Пусть D – полный чум. Элемент $a \in D$ называется *компактным*, если для любого направленного $X \subset D$

$$(a \preceq \sqcup X) \Rightarrow \exists x \in X (a \preceq x).$$

Обозначаем через D° множество компактных элементов полного чума D.

Множество $X \subset D$ называется *ограниченным*, если существует элемент $a \in D$ такой, что для всех $x \in X$ выполнено $x \preceq a$.

Определение 8.2. Полный чум D называется *доменом Скотта–Ершова*, если выполнены следующие два условия:

1) D алгебраическое, т. е. для любого его элемента u множество $\{d \in D |\ d -$ компактный и $d \preceq u\}$ является направленным и для него u является точной верхней гранью;

2) любое ограниченное подмножество в D имеет точную верхнюю грань.

Домен Скотта–Ершова называется *счетно основанным*, если D° счетное множество.

8.3.1. Топология Скотта

Пусть D – полный чум. Множество $U \subset D$ называется *открытым по Скотту*, если оно удовлетворяет следующим двум условиям:

1) $(x \in U) \wedge (x \preceq y) \Rightarrow (y \in U)$;

2) для любого направленного $X \subset D$

$$(\sqcup X \in U) \Rightarrow (X \cap U \neq \emptyset).$$

Совокупность открытых по Скотту множеств задает топологию на полном чуме D, которую называют *топологией Скотта*.

Домены Скотта, снабженные топологией Скотта, являются топологическими пространствами. Функция $f : D \longrightarrow D'$ между двумя доменами Скотта–Ершова *непрерывна*, если она непрерывна относительно топологий Скотта на D и D'.

Функции, непрерывные по Скотту, всегда монотонны, т. е. ($x \preceq y$) влечет ($f(x) \preceq f(y)$). Значит, чем больше информации в аргументе, тем больше ее и в значении. Это делает непрерывные по Скотту функции как-то похожими на вычислимые [40]. Следовательно, открывается путь построения общей теории вычислимых функций.

«Непрерывность является центральным столпом теории домена. Она служит средством качественного приближения к вычислимости. Другими словами, для большинства целей, чтобы обнаружить, что некоторая конструкция является вычислительно исполнимой, достаточно проверить, что она непрерывна; непрерывность проверяется легче, чем вычислимость» [50, p. 6].

8.3.2. Эффективный домен Скотта

Рассмотрим чум $< D, \preceq >$. Два элемента $d, d' \in D$ *совместимы*, символически $d \Uparrow d'$, если для них существует точная верхняя грань $d \sqcup d' = \sqcup\{d, d'\}$.

Определение 8.2. *Эффективный домен Скотта* – это тройка $< D, \preceq, e >$, где $< D, \preceq >$ – домен Скотта–Ершова, $e : \mathbb{N} \longrightarrow D^{\circ}$ нумерация множества компактных элементов, для которой рекурсивны следующие предикаты:

1) $e(n) \preceq e(m)$;
2) $e(n) \Uparrow e(m)$;
3) $e(n) = e(m) \sqcup e(m')$ [62, p. 3].

Пусть (D, α) и (E, β) – эффективные домены. Тогда непрерывное $f : D \longrightarrow E$ называется (α, β)-*эффективным*, если отношение $\beta(n) \preceq_E f(\alpha(m))$ рекурсивно перечислимо [63, p. 2].

8.3.3. Категория *EDom*

Категория эффективных доменов Скотта EDom состоит из эффективных доменов в качестве объектов и имеет морфизмами $(D, \alpha) \longrightarrow (E, \beta)$ соответствующие (α, β)-эффективные отображения, которые представляют вычислимые функции.

Категория *EDom* декартово замкнутая, но не имеет объекта натуральных чисел [89, p. 321].

Эффективный домен D ассоциируется с нумерацией α_D множества его компактных элементов D° и морфизмам $D \longrightarrow E$ соответствуют теоретико-множественные функции $D^{\circ} \longrightarrow E^{\circ}$. Это задает

вложение $EDom$ в категорию Ершова En. Посредством эффективных f_0-пространств Ю.Л. Ершов показал [65], что вложение будет полным.

Следовательно, с учетом (8.1), имеем полное вложение [89, p. 321]

$$\boxed{EDom \hookrightarrow RtM.} \tag{8.2}$$

Имеет место также полное вложение [89, p. 321]

$$\boxed{EDom \hookrightarrow \mathcal{E}\!f\!f.} \tag{8.3}$$

«Из этих вложений следует, что если мы изучаем эффективные домены, то не играет роли, описываем ли мы отображения как вычислимые в $EDom$, или эффективные в $\mathcal{E}\!f\!f$, или вычислимые в RtM (т. е. как естественные преобразования). Это объясняет, почему функционалы и операторы в классической теории рекурсии имеют разнообразные описания» [89, p. 321].

8.3.4. Синтетическая теория доменов

Мартин Хайлэнд, Джузеппе Розолини и другие предложили синтетическую теорию доменов в качестве такой теории множеств, в которой все множества являются доменами, а все теоретико-множественные функции являются «непрерывными», или «вычислимыми».

«Множества» при этом не понимаются в классическом смысле теории Кантора, а рассматриваются как объекты некоторого топоса с объектом натуральных чисел, поскольку следует найти аналог рекурсивным функциям.

8.4. Применение теории нумераций в программировании

Как уже говорилось, частичные комбинаторные алгебры применяются для нумерации алгоритмов. Если допустить, что задача по-

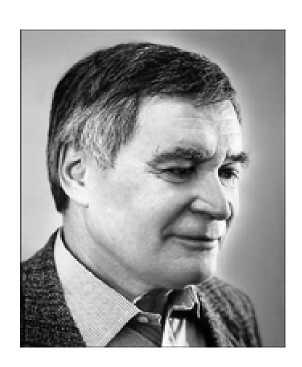
строения синтетического мира не является первостепенной по важности, то для решения каких других практически насущных задач теории вычислений могут оказаться полезными топосы реализуемости, построенные на обобщении теории нумераций, использующей не только натуральные числа?

В 1977 году Ю.Л. Ершов писал:

> Результаты теории нумераций оказались важными для прояснения ряда трудностей, возникающих в современной практике эксплуатации ЭВМ. Например, одной из важных задач современного программирования является задача эффективного построения по программе вычисления функции на одной машине программы вычисления той же функции на другой машине. Практическая реализация этих «переводов» («трансляций») для двух универсальных машин оказывается весьма сложной, а часто пока и не осуществленной. Чтобы понять, что эти трудности могут носить принципиальный характер, примем следующую интерпретацию рассматриваемой ситуации.
>
> Под «универсальной вычислительной машиной» будем понимать машину, вычисляющую некоторую двуместную функцию Ψ, универсальную для класса всех одноместных частично рекурсивных функций, а под «программой вычисления одноместной частично рекурсивной функции ϕ» будем понимать ее номер (один из ее номеров), т. е. такое число $n \in \mathbb{N}$, что $\phi = \lambda x.\Psi(n, x)$.
>
> Таким образом, если имеются две универсальные вычислительные машины (т. е. две универсальные функции Ψ_0, Ψ_1), то «проблема перевода» может быть сформулирована как проблема существования одноместной общерекурсивной функции f такой, что
>
> $$\lambda n \lambda x.\Psi_0(n, x) = \lambda n \lambda x.\Psi_0(n, x).$$
>
> Однако, как показывают исследования вычислимых нумераций, существуют такие универсальные функции Ψ_0, Ψ_1, что желаемой функции f не существует. Более того, существуют и такие Ψ_0, Ψ_1, что невозможен ни перевод с Ψ_0 на Ψ_1, ни наоборот.

Тем не менее, в классе так определенных универсальных машин существует «самая» универсальная (которую, по-видимому, только и стоит называть универсальной) в том смысле, что на язык этой машины может быть осуществлен перевод с любой другой универсальной машины. Если же рассматривать машины, которые вычисляют только общерекурсивные (всюду определенные) функции из некоторого достаточно богатого класса, то ситуация становится еще более плохой. Для любой такой машины существует другая машина, вычисляющая тот же класс функций, такая, что обе проблемы перевода неразрешимы. Указанный только что подход к разъяснению трудностей перевода может быть использован для определения понятия сложности класса функций (с помощью полурешетки вычислимых нумераций этого класса). Такое понятие сложности класса функций может, по-видимому, во многих вопросах быть более полезным, чем изучаемые сейчас различные понятия сложности отдельно взятых функций [21, с. 10].

8.5. Ю.Л. Ершов



Ю.Л. Ершов

Юрий Леонидович Ершов (род. 1 мая 1940 года, Новосибирск) – советский и российский математик.

Окончил Новосибирский государственный университет (НГУ) в 1963 году. Ученик академика АН СССР алгебраиста А.И. Мальцева. Кандидат наук (1963), доктор физико-математических наук (1965), профессор НГУ с 1968 года[2], в 1985–1993 годах – ректор НГУ.

В 1970 году Ю.Л. Ершов избирается членом-корреспондентом Академии наук СССР, с 1991 года – действительный член Российской академии наук. Директор Института математики СО РАН с 2003 по 2011 год.

В теории алгоритмов им была создана структурная теория

[2] В декабре 1966 г. автор данной книги, будучи студентом 2-го курса, увлекавшимся математической логикой, сдал успешно и досрочно Юрию Леонидовичу дисциплину «Теория алгоритмов», читавшуюся на 3-ем курсе.

нумераций, построена иерархия, которая теперь носит его имя – иерархия Ершова. По этой теме им в 1977 году опубликована монография «Теория нумераций» [21].

Одним из результатов теории нумераций было построение категории f_0-пространств [21, с. 358], с помощью которой может быть построена непротиворечивая модель бестипового λ-исчисления, во многих аспектах более удобная, чем топология Скотта [4].

Ю.Л. Ершов внес вклад в становление и развитие теории вычислимых (конструктивных) моделей. Теория конструктивных моделей связана с изучением моделей, допускающих вычислимые представления. Он построил, независимо от Д.С. Скотта, теорию денотационных семантик программ. «На основе этой теории им была построена теория вычислимости в допустимых надстройках над абстрактными моделями, были доказаны теоремы о существовании универсальных вычислимых отношений в этих надстройках, а также построена теория вычислимых отношений конечных типов. Эти результаты легли в основу нового подхода к созданию логических языков программирования – так называемого семантического программирования – и позволили взглянуть на вычислимость не через алгоритмическую реализацию, а через определимость в формальном языке, который допускает ясную семантику» [49, с. 7]. По этой теме написал монографию «Определимость и вычислимость» [20].

8.6. Ф. Малри

Филип С. Малри (Philip S. Mulry, род. в 1950 году) – профессор факультета компьютерных наук в Colgate University (США).



Ф.С. Малри

Ученые степени: бакалавр наук (Fairfield University, 1972); магистр искусств (1976, 1978), доктор философии (SUNY Buffalo, 1980).

Интересы: теория вычислений, дискретные структуры, языки программирования, объектно-ориентированное программирование, формальные методы, операционная и денотационная семантика, функциональные языки, такие как

Haskell, теория категорий.

Придумал рекурсивный топос, в котором все функции $F : N^N \to N$ являются вычислимыми.

Увлекается бегом на дистанции от 5 км до марафонской.

В 1989 году Малри отметил нарастающую тенденцию использования теории топосов в теории вычислений. Он писал:

> В прошлое десятилетие наблюдалась значительная деятельность, связанная с применением методов теории категорий к проблемам теории вычислений. Начиная с работ Ловера [79] и Ершова [66] арена обсуждения естественным образом переместилась от декартово закрытых категорий к топосам. Прежде всего следует отметить появление эффективного топоса [73] и рекурсивного топоса [91]. Одновременно, Скотт начал исследования в теории вычислений и денотационной семантике, используя топологические методы и методы теории решеток, которые частично также развились в изучение областей в топосах [95].

8.7. Д. Скотт



Д.С. Скотт

Дана Стюарт Скотт (Dana Stewart Scott, род. 11 октября 1932 года) – американский математик, известный работами в области математической логики и информатики.

Закончил Принстонский университет и Калифорнийский университет в Беркли. Научный руководитель Алонзо Чёрч.

Скотт сделал крупный вклад в развитие компьютерных наук, разработав математическую теорию вычислений. Работы в области компьютерных наук направлены на теорию и модели вычислений, включая разработку денотационной семантики языков программирования и математических основ вычислимости. Д. Скотт – один из создателей языка бесконечных формул.

Скоттом была разработана модель бестипового λ-исчисления, для чего была введена специальная топология, названная позже топологией Скотта, вначале на полных решетках, и впоследствии обобщенная до полных частично упорядоченных множеств. В рамках этой модели Скотт и развил теорию и модели вычислений, разработал принципы денотационной семантики языков программирования, углубил понятие о вычислимости.

Профессор Д. Скотт в 2009 году был награжден Золотой медалью Института математики им. С.Л. Соболева Сибирского отделения РАН за выдающийся вклад в развитие математики.

Заключение

Синтетические миры, в которых все функции вычислимы, исторически, в ходе развития теории вычислимости, не сводились только к поиску соответствующего топоса. Топос – это всего лишь среда, в которой реализуются основные конструкции, понимаемые как вычисления. В поиске описания таких сред разными авторами были найдены различные подходящие математические структуры. В итоге каждая из найденных сред была оформлена как категория, рассматриваемая как вселенная (синтетический мир), в той или иной мере пригодная для выявления и обсуждения сути вычислимости и эффективности [89].

Из найденных интересующих нас вселенных к настоящему времени помимо эффективного $\mathcal{E}\!f\!f$ и рекурсивного RtM топосов широко известными являются категория Ершова En, категории эффективных f_0-пространств Ершова и категория эффективных доменов Скотта $EDom$. Известны также категория частичных отношений эквивалентности $PER(A)$ и категория обобщенных перечисленных множеств.

Важный момент: топосы – это миры с интуиционистской логикой. В мирах Ершова и Скотта используется классическая логика. Она привычна для любого математика (и не только математика) и именно ей он пользуется, как правило, в своих выкладках.

В синтетическом мире эффективного топоса с его синтетической вычислимостью действует интуиционистская логика. Является ли эта логика препятствием к тому, чтобы воспринимать синтетический мир топоса $\mathcal{E}\!f\!f$ как мир *физически реальный*? Ведь в мире, который мы исторически считаем *реальным*, истинной считается двузначная классическая логика, на которой построено здание всей классической науки, и что самое важное – здание современной

физики, описывающей устройство окружающего нас Внешнего мира.

В монографиях [10], [11] продемонстрировано, как выглядит Внешняя реальность и как построена физика, основанные на интуиционистской логике. Соответствующая формальная теория использует в качестве интерпретации так называемый гладкий топос $\mathbf{Sets}^{\mathbf{L}^{op}}$, для которого проблема вычислимости всех функций $f : N \longrightarrow N$, насколько известно автору, не исследовалась. Было бы интересно найти интерпретацию интуиционистской физической теории в топосе, сочетающем универсальные свойства гладких топосов и эффективного топоса.

Поскольку интуиционистская физическая теория скорее всего более адекватно отражает психологию человеческого поведения, формирующего окружающий людей Внешний мир, то естественно ожидать от этого мира полной вычислимости. Интуиция исследователя в таком случае означала бы способность предугадывать результат очень сложных вычислений, требующих либо исторически больших отрезков времени, выходящих за пределы срока человеческой жизни (или критических для общественной системы), либо являющихся крайне технически и финансово затратных.

В настоящее время распространяется мнение [18, с. 7–8], что проблема сложных вычислений может быть решена с помощью квантовых компьютеров, способных за короткое время производить огромное число параллельных вычислений, осуществляемых посредством специальных квантовых алгоритмов:

> Модель квантовых вычислений – это некоторая математическая система, которая позволяет производить произвольные вычисления, причём уже показано, что для некоторых задач эта модель позволяет реализовать намного более эффективные алгоритмы, чем традиционная вычислительная модель, основанная на машине Тьюринга или лямбда-исчислении. Реализация этой новой вычислительной модели в «железе» не имеет фундаментальных преград, так что дело только за технологиями [18, с. 72].

Теория квантовых вычислений основывается на принципах квантовой механики, но оторванных от их физического содержания. Вычисляются состояния (квантовых) регистров абстрактного квантового компьютера, а не состояния электронов или атомов. Логика, проявляющаяся при анализе изменения квантовых состояний,

отлична от классической, но ее трудно отнести к непосредственно интуиционистской логике; за ней закрепилось название «*квантовая логика*» [14].

Возникает новая ситуация, касающаяся представления об абстрактном вычислении, отличная от той, о которой писал А.П. Ершов [19]. Кроме математиков-логиков, занимающихся теорией вычислимости, в основании которой, в значительной мере, лежит теория рекурсивных функций, и теоретиков-программистов, ищущих возможность «рассуждать о вычислимой функции раньше появления и независимо от наличия задающей ее программы»[1] [19], появились квантовые физики-математики, рассуждающие об универсальности языка квантовых вычислений, поскольку это язык эволюционирующей Вселенной [29], [80], [81]. Другими словами, теория квантовой вычислимости является теорией синтетической вычислимости.

Но помимо синтетичности теория квантовой вычислимости имеет еще одно важное преимущество по сравнению со всеми другими моделями вычислимости: для понимания сути квантовой теории требуется учитывать *сознание вычислителей*. И хотя квантовая механика далека от совершенства, поскольку в ней содержится лишь намек на то, что она учитывает сознание, и нет уверенности, что это действительно так, мы не можем не признать, что эта модель вычислимости заслуживает особого внимания и за ней стоит новое знание, касающееся природы вычислимого.

[1] Синтетическая вычислимость, основанная на топосах реализуемости и представленная в данной книге, развивается как раз математиками-логиками, работающими на факультетах компьютерных наук, т. е. общающимися с теоретиками-программистами.

Литература

[1] *Айзерман М.А. и др.* Логика, автоматы, алгоритмы. М., 1963.

[2] *Анисов А.М.* Абстрактная недетерминированная вычислимость // Credo New. 2009. №1.

[3] *Анисов А.М.* Время как вычислительный процесс // Замысел Бога в теориях физики и космологии. Время. СПб.: Изд-во С.-Петерб. ун-та, 2005. С. 53–71.

[4] *Барендрегт Х.* Ламбда-исчисление. Его синтаксис и семантика М.: Мир, 1985. 606 с.

[5] *Булос Дж., Джеффри Р.* Вычислимость и логика. М.: Мир, 1994. 396 с.

[6] *Вейль Г.* Математическое мышление. М., 1989.

[7] *Гейтинг А.* Интуиционизм. М., 1965.

[8] *Гливенко В.И.* Sur la logique de M. Brouwer // Bull. Acad. Sci. Belg, ser. 5. 1928. S. 225–228.

[9] *Гольдблатт Р.* Топосы. М.: Мир, 1983.

[10] *Гуц А.К.* Элементы теории времени. Изд. 2-е, доп. М.: УРСС, 2011. 376 с.

[11] *Гуц А.К.* Физика реальности. Омск: КАН, 2012. 424 с.

[12] *Гуц А.К.* Кибернетика: учебное пособие. Омск: Изд-во ОмГУ, 2014. 188 с.

[13] *Гуц А.К.* Математическая логика и теория алгоритмов. Изд. 3-е, испр. М.: УРСС, 2016. 128 с.

[14] *Гуц А.К.* Основы квантовой кибернетики. М.: УРСС, 2016.

[15] *Джонсон П.Т.* Теория топосов. М.: Наука, 1986.

[16] *Драгалин А.Г.* Конструктивная теория доказательств и нестандартный анализ. М.: Едиториал УРСС, 2003. 544 с.

[17] *Драгалин А.Г.* Математический интуиционизм. Введение в теорию доказательств. М., 1979.

[18] *Душкин Р.В.* Квантовые вычисления и функциональное программирование. М., 2014. 318 с.

[19] *Ершов А.П.* Абстрактная вычислимость в алгебраических системах // Алгоритмы в современной математике и ее приложениях. Часть II. Новосибирск: Изд-во ВЦ СО АН СССР, 1982. С. 194–223.

[20] *Ершов Ю.Л.* Определимость и вычислимость. Новосибирск: Научная книга, 1996. 286 с.

[21] *Ершов Ю.Л.* Теория нумераций. М.: Наука, 1977.

[22] *Ершов Ю.Л., Палютин Е.А.* Математическая логика. М.: Наука, 1979.

[23] *Ершов Ю.Л., Целищев В.В.* Алгоритмы и вычислимость в человеческом познании. Новосибирск: Изд-во СО РАН, 2012. 497 с.

[24] *Катленд Н.* Вычислимость. Введение в теорию рекурсивных функций. М.: Мир, 1983. 256 с.

[25] *Клини С.* Введение в метаматематику. М.: ИЛ, 1957.

[26] *Коганов А.В.* Возникновение противоречий в теории множеств Цермело–Френкеля при расширении базового языка рекурсивными функциями // Компьютерные исследования и моделирование. 2009. Т. 1, № 4. С. 367–380.

[27] *Колмогоров А.Н.* О принципе tertium non datur // Матем. сб. 1925. Т. 32, № 4. С. 646–667.

[28] *Кушнер Б.А.* Лекции по конструктивному математическому анализу. М.: Наука, 1973.

[29] *Ллойд С.* Программируя Вселенную. Квантовый компьютер и будущее науки. М.: Альпина нон-фикшн, 2013. 256 с.

[30] *Мальцев А.И.* Алгоритмы и рекурсивные функции. М.: ФМ, 1986. 368 с.

[31] *Манин Ю.И.* Вычислимое и невычислимое. М.: Советское радио, 1980. 128 с.

[32] *Марков А.А.* О логике конструктивной математики. М., 1970.

[33] *Марков А.А., Нагорный Н.М.* Теория алгорифмов. М.: Наука, Глав. ред. физ.-мат. лит-ры, 1984.

[34] *Мартин-Леф П.* Очерки по конструктивной математике. М.: Мир, 1975.

[35] *Мартыненко Б.К.* Языки и трансляции: учебное пособие. СПб.: Изд-во СПбГУ, 2004. 231 с.

[36] *Мендельсон Э.* Введение в математическую логику. М.: ФМ, 1971.

[37] *Новиков П.С.* Конструктивная математическая логика с точки зрения классической. М.: Наука, 1977. 328 с.

[38] *Роджерс Х.* Теория рекурсивных функций и эффективная вычислимость. М., 1972.

[39] *Светлов В.А.* Философия математики. Основные программы обоснования математики XX столетия: Учебное пособие. М.: КомКнига, 2006. 208 с.

[40] *Скотт Д.* Теория решеток, типы данных и семантика // Данные в языках программирования. М.: Мир, 1982. С. 25–53.

[41] Словарь философских терминов / Науч. ред. проф. В.Г. Кузнецова. М.: ИНФРА-М, 2007.

[42] Современный словарь иностранных слов. М.: Русский язык, 1992.

[43] *Тростников В.Н.* Конструктивные процессы в математике. М.: Наука, 1975.

[44] *Успенский В.А.* Лекции о вычислимых функциях. М.: ФМ, 1960.

[45] *Филд А., Харрисон П.* Функциональное программирование. М.: Мир, 1993. 637 с.

[46] *Хлызов Е.О.* Эффективный топос как синтетическая вселенная для теории вычислений // Математические структуры и моделирование. 2009. Вып. 19. С. 28-44.

[47] *Хювёнен Э., Сеппянен И.* Мир Лиспа. Т. 1. М.: Мир, 1990.

[48] *Шалак В.И.* Шейнфинкель и комбинаторная логика // Логические исследования. Вып. 15. М.: Наука, 2009. С. 247–265.

[49] Юрий Леонидович Ершов: Биобиблиографический указатель / Ред. С.С. Гончаров. Новосибирск: Изд-во Ин-та математики, 2010. 94 с.

[50] *Abramsky S., Jung A.* Domain Theory // Domain Theory in the Handbook of Logic in Computer Science. V. 3 / ed. S. Abramsky, Dov M. Gabbay, and T.S.E. Maibaum. Oxford: Clarendon Press, 1994. URL: http://www.cs.bham.ac.uk/ axj/pub/papers/handy1.pdf (corrected and expanded version).

[51] *Awodey S., Bauer A.* Sheaf Toposes for Realizability. 2004. URL: http://www.cs.cmu.edu/afs/cs/Web/People/andrej/papers/sheaves.pdf.

[52] *Bauer A.* An injection from $N^N \longrightarrow N$. – URL: http://math.andrej.com/wp-content/uploads/2011/06/injection.pdf.

[53] *Bauer A.* First steps in synthetic computability theory // Electr. Notes Theor. Comput. Sci. 2006. V. 155. P. 5–31.

[54] *Bauer A.* The Realizability Approach to Computable Analysis and Topology. Ph.D. Thesis. School of Computer Science Carnegie Mellon University Pittsburgh, 2000.

[55] *Beeson M.* Foundation of constructive mathematics. Sringer, 1985.

[56] *Bernadet A., Graham-Lengrand S.* A simple presentation of the effective topos. URL: http://arxiv.org/pdf/1307.3832v1.pdf.

[57] *Bernays P., Schönfinkel M.* Zum Entscheidungsproblem der mathematischen Logik // Mathematische Annalen. 1926. V. 99. S. 342–372.

[58] *Borceux F.* Handbook of Categorical Algebra. 3. Categories of Sheaves. Cambridge University Press, 1994.

[59] *Brouwer L.E.J.* Collected Works, Vol. 1 // Philosophy and Foundations of Mathematics. Amsterdam: Oxford, 1975.

[60] *Brouwer L.E.J.* Beweis, dass jede volle Funktion gleichmässig stetig ist (Proof, that every total function is uniformly continuous) // Nederl. Akad. Wetensch. Proc. 1924. V. 27. P.189–193.

[61] *Carboni A., Freyd P., Scedrov A.* A categorical approach to reability and polymorphic types / Lectures Notes in Computer Science. no. 298. Springer, 1988.

[62] *Dahlgren F.* A domain theoretic approach to effective distribution theory // U.U.D.M. Report, no. 37. Department of Mathematics Uppsala University, 2007. 66 p.

[63] *Dahlgren F.* Partial Continuous Functions and Admissible Domain Representations // Lecture Notes in Computer Science. 2007. V. 3988. P. 94–104.

[64] *Ershov Yu.L.* Theorie der Numerierungen. I // Z. Math. Logik Grundlag. Math. 1973. Bd. 19, Hf. 4. S. 289–388.

[65] *Eršov Ju.L.* Model C of partial continuous functions // Logic Colloquium 76. Amsterdam: North-Holland, 1977.

[66] *Eršov Ju.L.* La thexie des enumerations // Actes du Congress International des Mathematiciens. 1970. V. 1. P. 223–227.

[67] *Faber E.* Code-free recursion & realizability. Universiteit Utrecht February 24, 2014.

[68] *Frey J.* A Characterization of Realizability Toposes. arXiv:1404.6997v1 [math.CT]. 2014. URL: http://arxiv.org/pdf/1404.6997v1.pdf.

[69] *Freyd P.J., Scedrov A.* Categories, Allegories. Noth Holland, 1990.

[70] *Gentzen G.* Untersuchungen fiber das logische Schliessen // Math. Z. 1934. 39. S. 176–210, 405–431.

[71] *Heyting A.* Die formalen Regeln der intuitionistischen Logik, Sitzungsber Preuss // Akad. Wis, phys.-math. KL. 1930, S. 42–56.

[72] *Hyland J.M.E., Johnstone P.T., Pitts A.M.* Tripos theory // Math. Proc. Camb. Phil. Soc. 1980. V. 88. P. 205–232.

[73] *Hyland J.* The effective topos // The L.E.J. Brouwer Centenary Symposium, 1982. P. 165–216.

[74] *Hyland J.M.E., Robinson E.P., Rosolini G.* The Discrete Objects of the Effective Topos // Proc. London Math. Soc. 1990. V. 60. P. 1–36.

[75] *Kanda A.* Recursion theorems and effective domains // Annals of Pure and Applied Logic. 1988. V. 38. P. 289–300.

[76] *Kleene S.C.* On the interpretation of intuitionistic number theory // J. Symbolic Logic. 1945. V. 10. P. 109–124.

[77] *Kock A.* Synthetic Differential Geometry. Cambridge University Press, 1981.

[78] *Kupers A.P.M.* Real numbers in a topos and Brouwer's theorem. URL: http://jeroengoudsmit.com/seminars/topos-theory-2009/handout_sanderk.pdf.

[79] *Lawvere F.W.* Diagonal arguments and cartesian closed categories // Lecture Notes in Math. no. 92. Berlin: Springer, I969.

[80] *Lloyd S.* Computational capacity of the Universe // Physical Review Letters. 2002. V. 88, no. 237901. URL: http://arxiv.org/pdf/quant-ph/0110141.pdf.

[81] *Lloyd S.* The Universe as Quantum Computer. URL: http://arxiv.org/pdf/1312.4455v1.pdf.

[82] *Longley J.R.* Reability toposes and language semantics. Ph. D. thesis. University of Edinburg, 1994.

[83] *MacLane S., Moerdijk I.* Sheaves in geometry and logic: a first introduction to topos theory. Springer-Verlag, 1992.

[84] *Mazur S.* Computable analysis. Warszawa, 1963.

[85] *McLarty C.* Elementary Categories, Elementary Toposes. New York: Oxford university press, 1992.

[86] *Menni M.* Exact completions and toposes. Ph.D. Thesis, University of Edinburgh, 2000.

[87] *Moerdijk I., van Oosten J.* Topos Theory. Department of Mathematics, Utrecht University, 2007.

[88] *Moggi E.* Partial morphisms in categories of effective objects // Information and Computation. 1988. V. 76, Issues 2–3. P. 250–277.

[89] *Mulry P.S.* Some connections between models of computation // Contemporary Mathematics. 1989. V. 92. P. 317–324.

[90] *Mulry P.* Categoric approach to the theory of computation // Journal of Pure and Applied Algebra. 1989. V. 43. P. 293–305.

[91] *Mulry P.* Generalized Banach-Mazur functionals in the topos of recursive sets // Journal of Pure and Applied Algebra. 1982. V. 26. P. 71–83.

[92] *Mulry P.S.* The Topos of Recursive Sets. Ph.D. Thesis, State University of New York at Buffalo, 1980.

[93] *Pitts A.M.* The theory triposes. PhD. thesis. University of Cambridge, 1980.

[94] *Rogers H.* Theory of Recursive Functions and Effective Computability. New York: McGraw-Hill, 1967.

[95] *Rosolini G.* Continuity and effectiveness in topoi. Ph. D. thesis. University of Oxford, 1986.

[96] *Sallach D.L.* Recognition-Based Logic and Social Conflict: Toward a Topos Model // The 8th International Workshop on Agent-based Approach in Economic and Social Complex Systems. Tokyo: Shibaura Institute of Technology, 2013.

[97] *Schönfinkel M.* On the building blocks of mathematical logic // From Frege to Godel. Harvard University Press, 1967. P. 355–366.

[98] *Schönfinkel M.* Uber die Bausteine der mathematischen Logik // Mathematische Annalen. 1924. V. 92 S. 305–316.

[99] *Scott D.* Lectures on a Mathematical Theory Computation. Technical Monograph PRG-19. Oxford University, 1981.

[100] *Scott D.* Continuous lattices / in: Toposes, Algebraic Geometry and Logic. Lecture Notes in Math. no. 274. Berlin: Springer, 1972.

[101] *Streicher T.* Realizability. URL: http://www.mathematik.tu-darmstadt.de/streicher/REAL/REAL.pdf.

[102] *Simpsona A., Streicherb S.* Constructive toposes with countable sums as models of constructive set theory // Annals of Pure and Applied Logic. 2012. V. 163, Is. 10. P. 1419–1436.

[103] *van den Berg B.* The Herbrand topos // Mathematical Proceedings of the Cambridge Philosophical Society. 2013. V. 155, no. 2. P. 361–374.

[104] *van Oosten J.* Partial Combinatory Algebras of Functions // Notre Dame J. Formal Logic. 2011. V. 52, no. 4. P. 431–448.

[105] *van Oosten J.* Realizability. An Introduction to its Categorical Side // Studies in Logic and the Foundations of Mathematics. V. 152. Elsevier, 2008.

[106] *van Oosten J.* Realizability: An Historical Essay, 2000. URL: http://www.math.uu.nl/people/jvoosten/realizability/history.ps.gz.

[107] *Vermeeren S.* Realizability Toposes. Darwin College, Belgium, 2009. URL: http://www.stijnvermeeren.be/download/mathematics/essay.pdf.

Предметный указатель

Учебное издание

Александр Константинович **Гуц**

СИНТЕТИЧЕСКАЯ ВЫЧИСЛИМОСТЬ

Учебное пособие

Редактор *Д.С. Нерозник*
Технический редактор *Е.В. Лозовая*
Дизайн обложки *З.Н. Образова*

Подписано в печать 30.05.2016.
Формат 60 × 84 1/16. Печ. л. 9,5. Усл. печ. л. 8,8. Уч.-изд. л. 9,0.
Тираж 120 экз.

Издательство Омского государственного университета
644077, Омск-77, пр. Мира, 55а
Отпечатано на полиграфической базе ОмГУ
644077, Омск-77, пр. Мира, 55а